# V9003/V9103
# Three-Phase Energy Metering SoC

**Version:** **5.0**
**Release Date:** **May 24, 2019**

Specifications are subject to be changed without notice.

# Revision History

| Date | Version | Description |
|---|---|---|
| 2015.08.26 | 0.1 | Initial release |
| 2019.05.24 | 5.0 | Add V9103 |

# Introduction

V9003/V9103 is a low-power-consumption and high-performance three-phase metering SoC chip. It integrates the analog front end, energy metering module, enhanced 8052 MCU, RTC, Flash. V9003 integrates LCD driver. It can be used in a three-phase multi-functional energy meter.

# Contents

# Figure List

# Table List

# 1. Features

## General Features

- Single power supply 5V, and wide supply voltage operation 3.0 to 5.5V;
- Integrated analog front end, energy metering module, enhanced 8052 MCU, RTC, 64 kB Flash, and LCD driver;
- Energy metering accuracy, IEC 62053-21:2003, IEC 62053-22:2003, and IEC 62053-23:2003 supportive, less than 0.1% error on active energy metering over a dynamic range of 2000:1 at 25°C;
- High-performance dedicated RTC with 0.3 ppm crystal compensation accuracy over temperature variation;
- Multi-operational modes with low-power consumption options;
- Low-power consumption:
  - ✓ Typical current in Working Mode: 13 mA
  - ✓ Typical current in Sleep Mode: 8.8 μA
  - ✓ Sleep Mode, with LCD module on, but without a screen: 12 μA
  - ✓ Sleep Mode, current for RTC: 3 μA
- Operating temperature: -40~+85°C;
- Storage temperature: -55 °C ~ +150 °C
- Package:

  100-pin Low-profile Quad Flat Package (LQFP) (V9003).

  44-pin Low-profile Quad Flat Package (LQFP) (V9103).

## Energy Metering Features

- Eight high-performance oversampling Σ/ΔADC channels are available, including Voltage Channel UA/UB/UC, Current Channel IA/IB/IC/IN, and the M Channel for temperature, battery voltage and external signals measurement;
- Can be configured into 3-phase meters for both 3-wire and 4-wire service;
- Active energy, reactive energy, fundamental active energy, and fundamental reactive energy of every phase and on overall system can be metered;
- Positive/Negative active energy of every phase and on overall system can be metered;
- Active energy can be metered with error less than 0.1% within dynamic range of 2000:1;
- Reactive energy can be metered with error less than 0.1% within dynamic range of 1000:1;
- Current RMS, voltage RMS, and fundamental current/voltage RMS can be metered;
- DC component in every channel can be metered;
- Power factor angles of every phase are available;
- Frequency and phase measurement is available;
- No-load detection is available, and no-load detection threshold is configurable;

- Current through phase wires and neutral wire is monitored continuously to detect fault of power distribution system;
- Active power gain can be calibrated in 3 segments;
- Phase error is configurable over a range of ±2.8 degrees, which can be compensated in 5 segments;
- Two configurable CF pulse outputs and CF interrupts are available;
- Meter up to the 31st harmonic.

## MCU Features

- High performance 8-bit 8052 compatible MCU core;
- 64-KB Flash memory supporting ISP (In System Programming) and IAP (In Application Programming), with write protection and encryption function;
- 4-KB SRAM memory;
- Debugging via JTAG interfaces in real-time;
- Integrated 160 segments (40×4) LCD driver; internal resistor ladder or charge pump for LCD waveform generation, and 3.0V/3.3V/3.5V driving capability;
- Up to 6 UARTs. One UART supporting IR communication;
- High-performance RTC with a crystal compensation accuracy of 0.3 ppm/s over temperature variation; integrated temperature measurement circuit with 1°C precision;
- 2 timers with similar function of MSP430;
- Sufficient timer resources, 5 hardware timers, two of them with similar function of timers in MSP430;
- Integrated watchdog timer (WDT), unable to be turned off, able to generate a reset signal without feeding in 1.5 seconds. WDT reset is masked when debugging;
- Configurable GPIO ports. Providing port interrupts;
- Different work modes and various wake-up events for different power consumption requirements

## Analog Circuit Features

- Embedded Programmable Gain Amplifiers (PGAs) providing a direct and flexible sensor interface;
- 8 independent high-performance oversampling Σ/Δ ADCs, enabling metering error of active energy to be less than 0.1% within dynamic range of 2000:1; One of ADCs can be configured to measure any voltage signal;
- Low-power consumption mode, ADCs supporting metering at 1/4 and 1/2 clock rate;
- Power supply monitor; two on-chip regulators generating programmable 2.5 V and 3.3 V voltages respectively;
- Power-down monitoring and battery voltage measurement circuit, enabling swift switch between regular and battery power supply;
- Integrated oscillator circuit and PLL; Only an external 32768-Hz crystal is needed;
- Internal 1.185-V reference with a typical temperature drift of 20 ppm/°C;

- Power-on Reset and Brown-out Reset, even in power-down mode.

# Functional Block Diagram

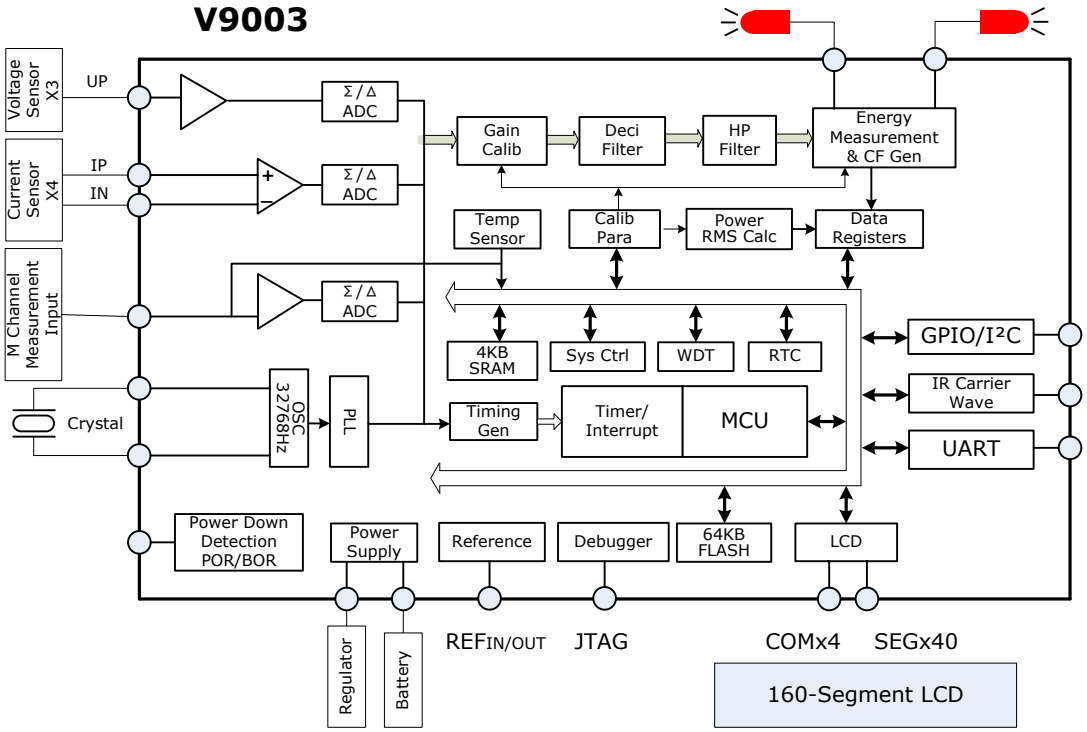



**Figure 1-1 V9003 Functional Block Diagram**

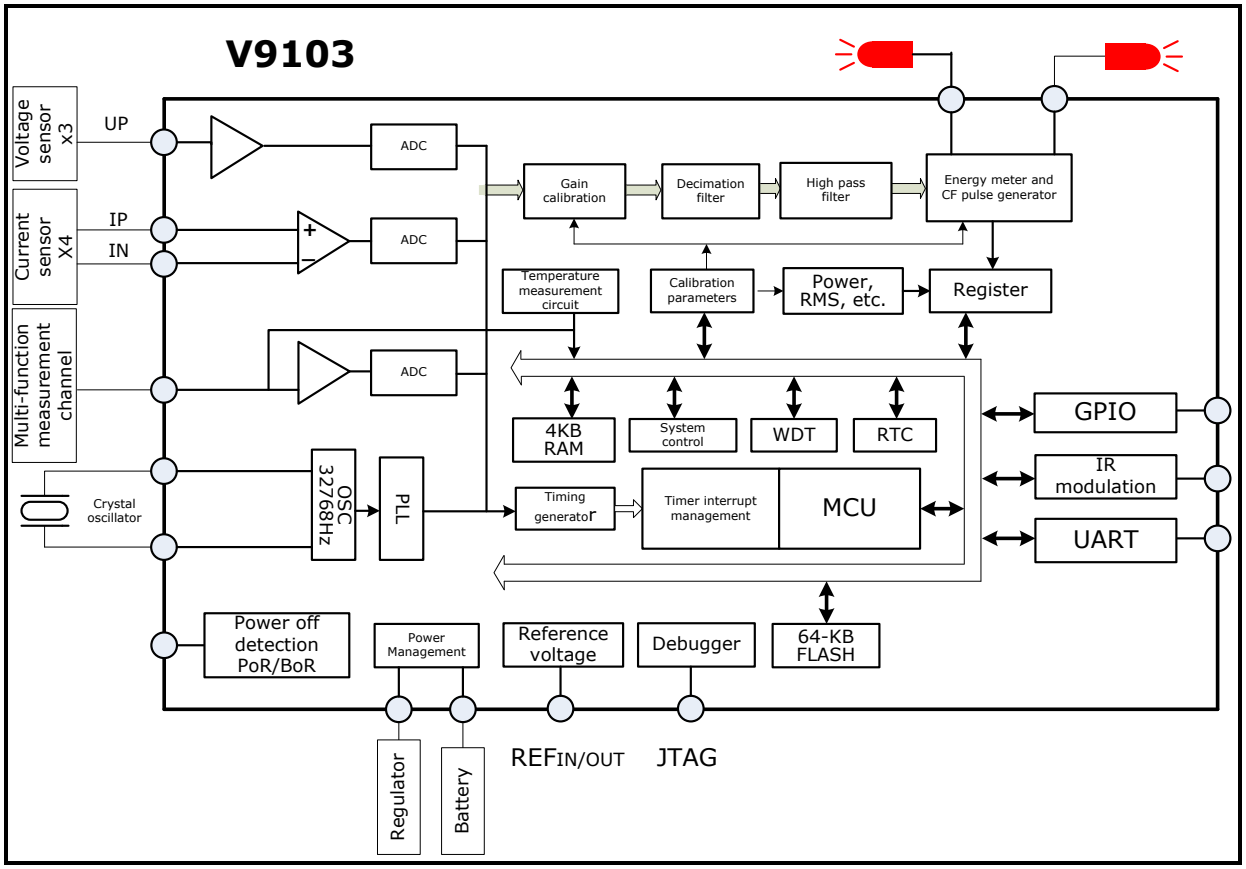
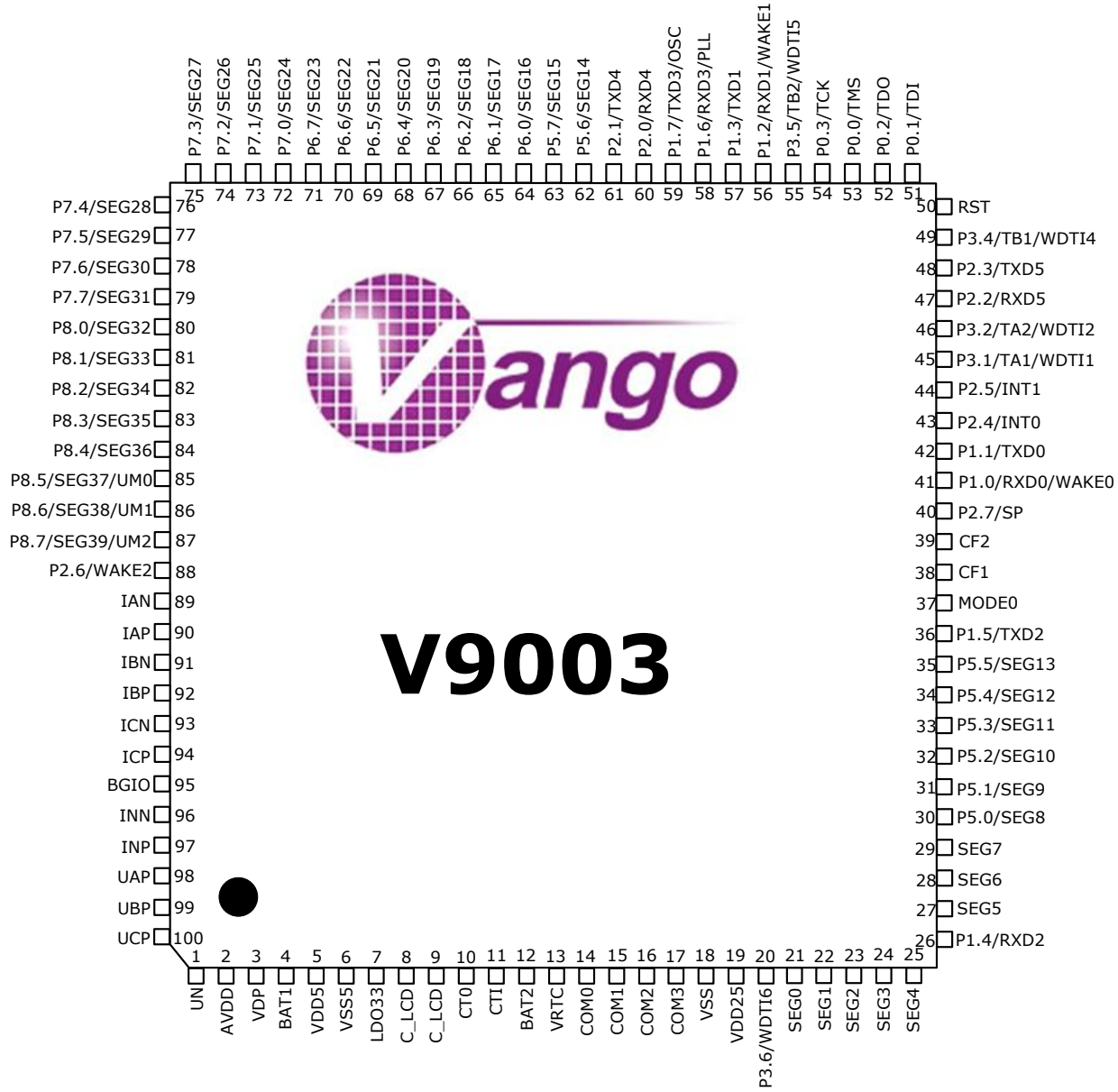
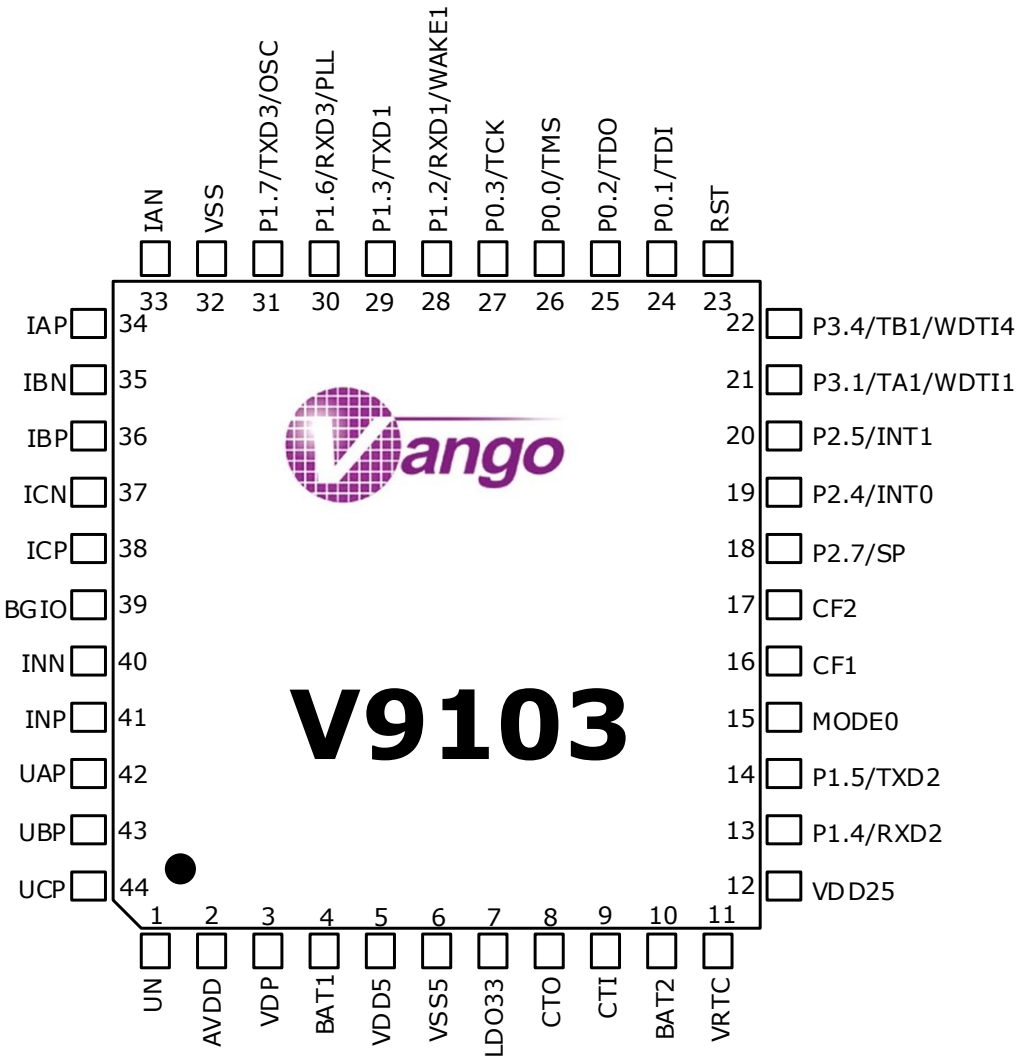



**Figure 1-2 V9103 Functional Block Diagram**

# Pin Descriptions






# Functional Block Diagram

| V9003 | V9103 | Mnemonic | Input/Output | Description |
|---|---|---|---|---|
| 1 | 1 | UN | Input | Negative input for voltage channels |
| 2 | 2 | AVDD | Input | 3.3V analog power input |
| 3 | 3 | VDP | Input | Power-down detection signal input. When the input level on the pin VDP is lower than 1.0V, the 5V main supply is powered down, even the power supply is switched to battery supply, which depends on the resistance ratio on the pin VDP and the voltage drop on the two diodes in the power-down detection circuit. |

| | | | | | |
|---|---|---|---|---|---|
| 4 | 4 | BAT1 | Input | To measure the main battery voltage or external voltage signal/ For battery discharge, to prevent the battery from passivation | |
| 5 | 5 | VDD5 | Input | 5V power input | |
| 6 | 6 | VSS5 | Input | 5V ground input | |
| 7 | 7 | LDO33 | Output | 3.3 V analog power supply output. This pin is connected internally to 3.3 V LDO. It should be connected a 0.1uF and 10uF de-couple capacitor to the ground, and then connected to AVDD. | |
| 8 | | C_LCD | Output | Capacitor pin for LCD charge pump driver | |
| 9 | | C_LCD | Output | Capacitor pin for LCD charge pump driver | |
| 10 | 8 | CTO | Output | Crystal output | Only a 32768Hz crystal is needed to connect externally to the chip. |
| 11 | 9 | CTI | Input | Crystal input | |
| 12 | 10 | BAT2 | Input | To measure the RTC battery voltage/ For battery discharge, to prevent the battery from passivation | |
| 13 | 11 | VRTC | Input | To power RTC with battery (2.3~3.0V, and 2.7V is recommended). This pin can be powered by LDO33 or batteries. | |
| 14 | | COM0 | Input/Output | Only for common output 0 | |
| 15 | | COM1 | Input/Output | Only for common output 1 | |
| 16 | | COM2 | Input/Output | Only for common output 2 | |
| 17 | | COM3 | Input/Output | Only for common output 3 | |
| 18 | | VSS | Output | Ground reference for the digital circuit | |
| 19 | 12 | VDD25 | Output | 2.5 V digital power supply output. This pin is connected internally to 2.5 V LDO. It should be connected a 0.1uF and 10uF de-couple capacitor to the ground. | |
| 20 | | P3.6/WDTI 6 | Input/Output | General-purpose Digital I/O Port/ P3 Interrupt 6, external interrupt source to trigger WDTI | |

| | | | | |
|---|---|---|---|---|
| | | | | interrupt |
| 21~25 | | SEG0~4 | Output | LCD segment output |
| 26 | 13 | P1.4/RXD2 | Input/Output | General-purpose Digital I/O Port/ Receiver data input of UART2 (asynchronous). When it is configured for IR communication, it should be connected externally to a demodulator. |
| 27~29 | | SEG5~7 | Output | LCD segment output |
| 30~35 | | P5.0~P5.5/ SEG8~SEG 13 | Input/Output | General-purpose Digital I/O Port/ LCD segment output |
| 36 | 14 | P1.5/TXD2 | Input/Output | General-purpose Digital I/O Port/ Transmitter data output of UART2, can be configured to transmit 38kHz carrier wave. |
| 37 | 15 | MODE0 | Input | To configure the work mode.<br>0, debugging; 1, normal |
| 38 | 16 | CF1 | Output | Active energy CF pulse output |
| 39 | 17 | CF2 | Output | Reactive energy CF pulse output |
| 40 | 18 | P2.7/SP | Input/Output | General-purpose Digital I/O Port/ Second pulse output |
| 41 | | P1.0/RXD0/ WAKE0 | Input/Output | General-purpose Digital I/O Port/ Receiver data input of UART0/ IO wake-up input, to wake up the system from Sleep |
| 42 | | P1.1/TXD0 | Input/Output | General-purpose Digital I/O Port/ Transmitter data output of UART0 |
| 43 | 19 | P2.4/INT0 | Input/Output | General-purpose Digital I/O Port/ IO interrupt 0 |
| 44 | 20 | P2.5/INT1 | Input/Output | General-purpose Digital I/O Port/ IO interrupt 1 |
| 45 | 21 | P3.1/TA1/ WDTI1 | Input/Output | General-purpose Digital I/O Port / Timer A port 1, to input or output the signals for Timer A Compare/Capture Module 1/ P3 Interrupt 1, external interrupt source to trigger WDTI interrupt |
| 46 | | P3.2/TA2/ WDTI2 | Input/Output | General-purpose Digital I/O Port / Timer A port 2, to input or output the signals for Timer A Compare/Capture Module 2/ P3 Interrupt 2, |

| | | | | |
|---|---|---|---|---|
| | | | | external interrupt source to trigger WDTI interrupt |
| 47 | | P2.2/RXD5 | Input/Output | General-purpose Digital I/O Port / Receiver data input of UART5 |
| 48 | | P2.3/TXD5 | Input/Output | General-purpose Digital I/O Port / Transmitter data output of UART5 |
| 49 | 22 | P3.4/TB1/ WDTI4 | Input/Output | General-purpose Digital I/O Port / Timer B port 1, to input or output the signals for Timer B Compare/Capture Module 1/ P3 Interrupt 4, external interrupt source to trigger WDTI interrupt |
| 50 | 23 | RST | Input | Reset input, active low |
| 51 | 24 | P0.1/TDI | Input/Output | General-purpose Digital I/O Port /JTAG port, test data input |
| 52 | 25 | P0.2/TDO | Input/Output | General-purpose Digital I/O Port /JTAG port, test data output |
| 53 | 26 | P0.0/TMS | Input/Output | General-purpose Digital I/O Port /JTAG port, test mode select |
| 54 | 27 | P0.3/TCK | Input/Output | General-purpose Digital I/O Port /JTAG port, test clock input |
| 55 | | P3.5/TB2/ WDTI5 | Input/Output | General-purpose Digital I/O Port / Timer B port 2, to input or output the signals for Timer B Compare/Capture Module 2/ P3 Interrupt 5, external interrupt source to trigger WDTI interrupt |
| 56 | 28 | P1.2/RXD1/ WAKE1 | Input/Output | General-purpose Digital I/O Port / Receiver data input of UART1/ IO wake-up input, to wake up the system from Sleep |
| 57 | 29 | P1.3/TXD1 | Input/Output | General-purpose Digital I/O Port / Transmitter data output of UART1 |
| 58 | 30 | P1.6/RXD3/ PLL | Input/Output | General-purpose Digital I/O Port / Receiver data input of UART3/PLL clock output<br>When P16FSEL=0 (bit6 of P1FSEL, 0x1A15) and P16FNC=1 (bit6 of P1FS, 0x1A14), this pin is used for receiver data input of UART3; |

| | | | | |
|---|---|---|---|---|
| | | | | When P16FSEL=1 and P16FNC=1, this pin is used to output PLL clock. |
| 59 | 31 | P1.7/TXD3/ OSC | Input/Output | General-purpose Digital I/O Port / Transmitter data output of UART3/OSC clock output<br>When P17FSEL=0 (bit7 of P1FSEL, 0x1A15) and P17FNC=1 (bit7 of P1FS, 0x1A14), this pin is used to for transmitter data output of UART3;<br>When P17SEL=1 and P17FNC=1, this pin is used to output OSC clock. |
| | 32 | VSS | Output | Ground reference for the digital circuit |
| 60 | | P2.0/RXD4 | Input/Output | General-purpose Digital I/O Port / Receiver data input of UART4 |
| 61 | | P2.1/TXD4 | Input/Output | General-purpose Digital I/O Port / Transmitter data output of UART4 |
| 62~84 | | P5.6~P8.4/ SEG14~36 | Input/Output | General-purpose Digital I/O Port / LCD segment output |
| 85 | | P8.5/SEG3 7/UM0 | Input/Output | General-purpose Digital I/O Port / LCD segment output/ Input pin for M Channel, to measure main battery voltage or external voltage signals |
| 86 | | P8.6/SEG3 8/UM1 | Input/Output | General-purpose Digital I/O Port / LCD segment output/ Input pin for M Channel, to measure main battery voltage or external voltage signals |
| 87 | | P8.7/SEG3 9/UM2 | Input/Output | General-purpose Digital I/O Port / LCD segment output/ Input pin for M Channel, to measure main battery voltage or external voltage signals |
| 88 | | P2.6/WAKE 2 | Input/Output | General-purpose Digital I/O Port / IO wake-up input, to wake up the system from Sleep or Deep Sleep |
| 89, 90 | 33, 34 | IAN, IAP | Input | Analog input for current channel of Phase A (Channel IA) |
| 91, 92 | 35, 36 | IBN, IBP | Input | Analog input for current channel of Phase B (Channel IB) |

| | | | | |
|---|---|---|---|---|
| 93, 94 | 37, 38 | ICN, ICP | Input | Analog input for current channel of Phase C (Channel IC) |
| 95 | 39 | BGIO | Input/Output | Voltage reference I/O port. This pin should be decoupled with a 1μF capacitor. |
| 96, 97 | 40, 41 | INN, INP | Input | Analog input for Channel IN |
| 98 | 42 | UAP | Input | Positive input for voltage channel of Phase A (Channel UA) |
| 99 | 43 | UBP | Input | Positive input for voltage channel of Phase B (Channel UB) |
| 100 | 44 | UCP | Input | Positive input for voltage channel of Phase C (Channel UC) |

# 2. Electrical Characteristics

## 2.1. Absolute Parameters

**Table 2-1 Absolute Parameters**

| Parameter | Min | Typ | Max | Unit | Description |
|---|---|---|---|---|---|
| Digital IO, Output | | | | | |
| Output High Voltage $V_{OH}$ | 2.4 | | | V | 8mA current cannot damage the chip in a short period of time; Long duration of 4mA current or above may cause damage to the chip.<br>The pin LDO33 is internally connected to digital 3.3VLDO. The current dissipated on IOs should not be over the maximum driving capacity of 3.3V LDO regulator. |
| $I_{SOURCE}$ | | 4 | 8 | mA | |
| Output Low Voltage $V_{OL}$ | | | 0.4 | V | |
| $I_{SINK}$ | | 4 | 8 | mA | |
| Digital IO, Input | | | | | |
| Input High Voltage $V_{INH}$ | 2.0 | | | V | |
| Input Low Voltage $V_{INL}$ | | | 0.4 | V | |

| | | | | | |
|---|---|---|---|---|---|
| Power Supply Input | | | | | |
| VDD5 | 3.0 | 5.0 | 9.0 | V | |
| VRTC | 2.3 | 2.7 | 3.0 | V | |
| LDO33 Output | | | | | |
| Voltage | | 3.3 | | V | |
| Current | | | 30 | mA | |
| LDO25 Output | | | | | |
| Voltage | 2.1 | 2.5 | 2.8 | V | programmable |
| Current | | | 25 | mA | |
| Power Consumption | | | | | |
| Normal Mode | | 13 | | mA | |
| Sleep Mode | | 8.8 | | μA | The crystal oscillator circuit, RTC, power monitor circuit, reset control module, and regulator work normally, and RAM memory holds the data. |

## 2.2. Analog Circuits Parameters

**Table 2-2 Analog Circuit Parameters**

| Parameter | Min. | Typ. | Max. | Unit | Descriptions |
|---|---|---|---|---|---|
| Analog Inputs | | | | | |
| Max Signal Levels | | | ±200 | mV | |
| Bandwidth (-3dB) | | 3.2 | | KHz | |
| ADC Performance | | | | | |
| DC Bias | | | 10 | mV | |
| Effective Bits | | 16 | | BIT | |
| On-chip Reference | | | | | |
| Reference Error | -18 | | 18 | mV | |
| Power Supply Rejection ratio | | 80 | | dB | |
| Temperature Coefficient | | 20 | | ppm/°C | |

| | | | | | |
|---|---|---|---|---|---|
| Output Voltage | | 1.185 | | V | |
| Power On Reset (POR) | | | | | |
| Detection Threshold on LDO25 | 1.7 | 1.8 | 1.9 | V | |
| VDP | | | | | |
| Max Signal Levels | 0 | | VDD | V | |
| Input Impedance (DC) | | 1.5 | | MΩ | |
| Low Detection Threshold for PWRDN | | 1.0 | | V | |
| Low Detection Threshold for PWRUP | | 1.1 | | V | |

# 2.3. Energy Metering Circuits Parameters

**Table 2-3 Energy Metering Circuit Parameters**

| Parameter | Typ. | Unit | Description |
|---|---|---|---|
| Phase Error Between Channels | | | |
| PF=0.8 Capacitive | ±0.05 | Degree | |
| PF=0.5 Inductive | ±0.05 | Degree | |
| Active Energy Metering Error | 0.1 | % | Dynamic range of 2000: 1@25°C |
| Active Energy Metering Bandwidth | 6.4 | kHz | |
| Reactive Energy Metering Error | 0.1 | % | Dynamic range of 1000: 1@25°C |
| Reactive Energy Metering Bandwidth | 6.4 | kHz | |
| Fundamental Active Energy Metering Error | 0.1 | % | Dynamic range of 2000: 1@25°C |
| Fundamental Active Energy Metering Bandwidth | 100 | Hz | |
| Fundamental Reactive Energy Metering Error | 0.1 | % | Dynamic range of 1000: 1@25°C |
| Fundamental Reactive Energy Metering Bandwidth | 100 | Hz | |
| Voltage RMS Metering Error | 1 | % | Dynamic range of 1000: 1@25°C |

| | | | |
|---|---|---|---|
| Voltage RMS Metering Bandwidth | 3.2 | kHz | |
| Current RMS Metering Error | 1 | % | Dynamic range of 1000: 1@25°C |
| Current RMS Metering Bandwidth | 3.2 | kHz | |
| CF Pulse Output | | | |
| Max. Output Freq. | 3.2 | kHz | |
| Duty Cycle | 50% | | When active high pulse width <80ms. |
| Active High Pulse Width | 80 | ms | |

# 2.4. Timing Parameters

**Table 2-4 Timing Parameters**

| Parameter | Typ. | Unit | Description |
|---|---|---|---|
| CPU Clock Frequency | 6553600 | Hz | Turn on PLL, and PLL output is used as the CPU clock source. |
| | 32768 | Hz | Default clock frequency after reset. |
| Crystal Frequency | 32768 | Hz | OSC clock |

# 3. 8052 MCU Core Architecture

The V9003/V9103 contains memory modules as follows:

- 64 Kbytes of on-chip Flash;
- 256 bytes of internal RAM of MCU (IRAM);
- 4096 bytes of internal extended RAM (XRAM);
- 60 KBytes external register addresses.

The IRAM shares the upper 128 bytes of its address space with Special Function Registers (SFRs). The XRAM and external register addresses are mapped into the data memory space. The Flash space is mapped into the program memory space.

## 3.1. IRAM (INTERNAL RAM of MCU) and SFRs (Special Function Registers)

### 3.1.1. IRAM

MCU stack, register bank and upper 128 bytes on-chip memory accessible through indirect addressing are implemented in a 256 bytes synchronous static RAM block. As long as the output voltage of LDO25 is higher than 1.7V, the data stored in this RAM will not be lost, and reset cannot affect it. After reset, the IRAM gets into normal power consumption mode. When in Sleep or Deep Sleep, it gets into low power consumption mode. The address range is 0x00~0xFF.

The lower 128 bytes are organized as shown in Figure 3-1. The lowest 32 bytes (0x00~0x1F) form four banks with eight registers each (R0~R7). Users can configure bit4 (RS1) and bit3 (RS0) of the register PSW SFR (0xD0, Program Status Word SFR) to select the register bank to be used. The next sixteen bytes (0x20~0x2F) form the bit address area with bit addresses from 0x00 to 0x7F, which is bit addressable. All of the bytes in the lower 128 bytes are accessible through direct or indirect addressing.

The SFRs and upper 128 bytes of IRAM share the same address range (0x80~0xFF). However, the actual address space is separate and is differentiated by the type of addressing. SFRs can be accessed via direct addressing, while the upper 128 bytes IRAM can be accessed via indirect addressing.

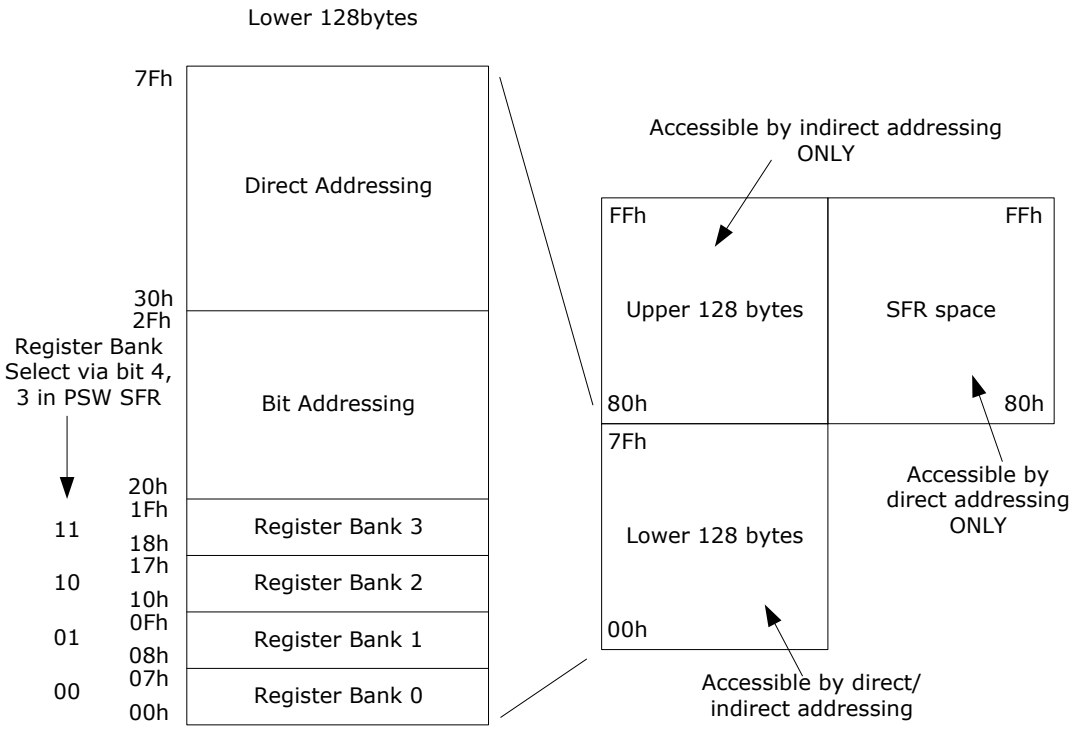



**Figure 3-1 IRAM and SFR**

## 3.1.2. SFRs

SFRs are categorized into internal SFRs in CPU and system SFRs.

Table 3-1 lists all SFRs.

**Table 3-1 SFRs**

| Register | bit 7 | bit 6 | bit 5 | bit 4 | bit 3 | Bit 2 | bit 1 | bit 0 | Address |
|---|---|---|---|---|---|---|---|---|---|
| Reserved | | | | | | | | | 80h |
| SP | - | - | - | - | - | - | - | - | 81h |
| DPL0 | - | - | - | - | - | - | - | - | 82h |
| DPH0 | - | - | - | - | - | - | - | - | 83h |
| DPL1 | - | - | - | - | - | - | - | - | 84h |
| DPH1 | - | - | - | - | - | - | - | - | 85h |
| DPS | 0 | 0 | 0 | 0 | 0 | 0 | 0 | SEL | 86h |
| PCON | SMOD0 | - | 1 | 1 | GF1 | GF0 | STOP | IDLE | 87h |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| TCON | TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 | 88h |
| TMOD | GATE1 | C/T1 | T1M1 | T1M0 | GATE0 | C/T0 | T0M1 | T0M0 | 89h |
| TL0 | - | - | - | - | - | - | - | - | 8Ah |
| TL1 | - | - | - | - | - | - | - | - | 8Bh |
| TH0 | - | - | - | - | - | - | - | - | 8Ch |
| TH1 | - | - | - | - | - | - | - | - | 8Dh |
| CKCON | - | - | T2M | T1M | T0M | MD2 | MD1 | MD0 | 8Eh |
| SPC_FNC | 0 | 0 | 0 | 0 | 0 | 0 | 0 | WRS | 8Fh |
| Reserved | | | | | | | | | 90h |
| EXIF | IE5 | IE4 | IE3 | IE2 | 1 | 0 | 0 | 0 | 91h |
| Reserved | - | - | - | - | - | - | - | - | 92h |
| Reserved | | | | | | | | | 93h |
| Reserved | | | | | | | | | 94h |
| Reserved | | | | | | | | | 95h |
| Reserved | | | | | | | | | 96h |
| Reserved | | | | | | | | | 97h |
| SCON0 | SM00 | SM10 | SM20 | REN0 | TB80 | RB80 | TI0 | RI0 | 98h |
| SBUF0 | - | - | - | - | - | - | - | - | 99h |
| Reserved | | | | | | | | | 9Ah |
| Reserved | | | | | | | | | 9Bh |
| Reserved | | | | | | | | | 9Ch |
| Reserved | | | | | | | | | 9Dh |
| Reserved | | | | | | | | | 9Eh |
| Reserved | | | | | | | | | 9Fh |
| Reserved | | | | | | | | | A1h |
| Reserved | | | | | | | | | A2h |
| IE | EA | ES1 | ET2 | ES0 | ET1 | EX1 | ET0 | EX0 | A8h |
| IP | 1 | PS1 | PT2 | PS0 | PT1 | PX1 | PT0 | PX0 | B8h |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| SCON1 | SM01 | SM11 | SM21 | REN1 | TB81 | RB81 | TI1 | RI1 | C0h |
| SBUF1 | - | - | - | - | - | - | - | - | C1h |
| T2CON | TF2 | EXF2 | RCLK | TCLK | EXEN2 | TR2 | C/T2 | CP/RL2 | C8h |
| Reserved | | | | | | | | | C9h |
| RCAP2L | - | - | - | - | - | - | - | - | CAh |
| RCAP2H | - | - | - | - | - | - | - | - | CBh |
| TL2 | - | - | - | - | - | - | - | - | CCh |
| TH2 | - | - | - | - | - | - | - | - | CDh |
| Reserved | - | - | - | - | - | - | - | - | CEh |
| Reserved | - | - | - | - | - | - | - | - | CFh |
| PSW | CY | AC | F0 | RS1 | RS0 | OV | F1 | P | D0h |
| EICON | SMOD1 | 1 | EPFI | PFI | WDTI | 0 | 0 | 0 | D8h |
| Reserved | - | - | - | - | - | - | - | - | D9h |
| ACC | - | - | - | - | - | - | - | - | E0h |
| EIE | 1 | 1 | 1 | EWDI | EX5 | EX4 | EX3 | EX2 | E8h |
| B | - | - | - | - | - | - | - | - | F0h |
| EIP | 1 | 1 | 1 | PWDI | PX5 | PX4 | PX3 | PX2 | F8h |

## 3.2. Data Memory Space

The addresses of 4096-Byte XRAM, SRAM for external registers and energy metering module, and addresses for configuring registers, are mapped into the data memory space (Figure 3-2).

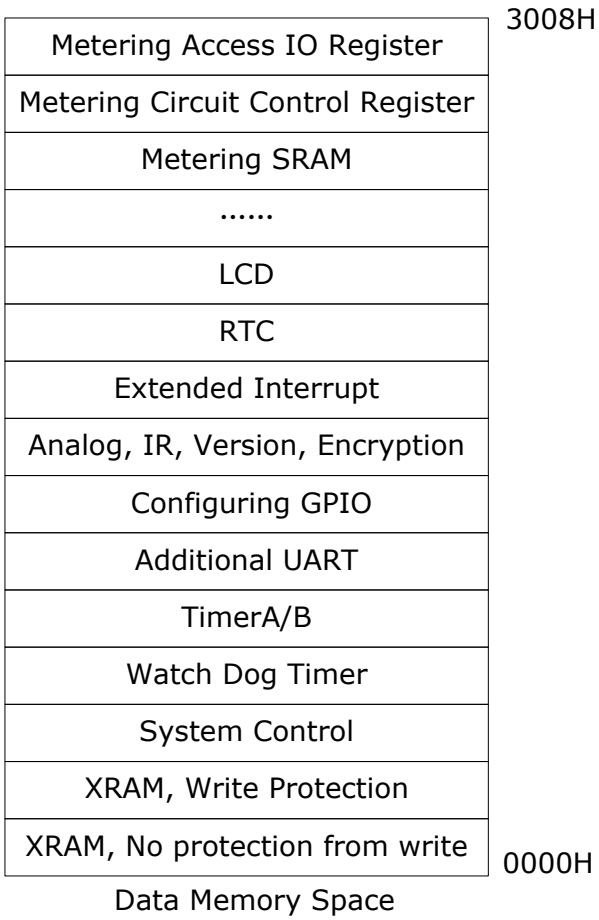



**Figure 3-2 Data Memory Space**

**Table 3-2 Address Distribution of XDATA**

| No. | Start Address | Bit Width (BIT) | Width | Direct Access of MCU? | Description | Reset by* |
|---|---|---|---|---|---|---|
| 1 | 0x0000 | 8 | 4096-BYTE | Yes | On-chip 4-kByte SRAM | No |
| 2 | 0x1200 | 8 | 11-BYTE | Yes | System status control registers | Level 1 |
| 3 | 0x1300 | 8 | 2-BYTE | Yes | WDT control registers | Level 2/1 |
| 4 | 0x1400 | 8 | 16-BYTE | Yes | Timer A control registers | Level 3/2/1 |
| 5 | 0x1500 | 8 | 16-BYTE | Yes | Timer B control registers | Level 3/2/1 |
| 6 | 0x1600 | 8 | 8-BYTE | Yes | UART2 control registers | Level 3/2/1 |
| 7 | 0x1700 | 8 | 8-BYTE | Yes | UART3 control registers | Level 3/2/1 |
| 8 | 0x1800 | 8 | 8-BYTE | Yes | UART4 control registers | Level 3/2/1 |
| 9 | 0x1900 | 8 | 8-BYTE | Yes | UART5 control registers | Level 3/2/1 |
| 10 | 0x1A00 | 8 | 53-BYTE | Yes | GPIO control registers | Level 3/2/1 |
| 11 | 0x1B00 | 8 | 38-BYTE | Yes | Control register bank | Level 3/2/1 |
| 12 | 0x1C00 | 8 | 25-BYTE | Yes | Interrupt control registers | Level 3/2/1 |
| 13 | 0x1D00 | 8 | 14-BYTE | Yes | RTC control registers | Registers located at 0x1D00~0x1D05 and 0x1D0B can be reset only when RTCRSTn=1, other RTC-related registers can be reset by Level 1. |
| 14 | 0x1F00 | 8 | 21-BYTE | Yes | LCD control registers | Level 1 |
| 15 | 0x2000 | 32 | N/A | No | Metering Configuration Registers, MTPARA0~3 | No |
| 16 | 0x2800 | 32 | 10-WORD | No | Metering control registers | No |
| 17 | 0x3000 | 8 | 9-BYTE | Yes | Metering access registers | Level 3/2/1 |

* Reset level, see "Reset" for details.

The address range of XRAM is 0x0000~0x0FFF. 0x0000~ 0x0F7F has free access and 0x0F80~0x0FFF is protected by XRAM Write Protection Register (0x1B20). Write of 0x0F80~0x0FFF is permitted when protection is disabled. And 0x0F80~0x0FFF is read only if write protection is restored.

XRAM is implemented by a 4096-byte synchronous static RAM block. As long as the output voltage of LDO25 is higher than 1.7V, the data stored in this RAM will not be lost, and reset cannot affect it. After reset, the XRAM gets into normal power consumption mode. When in Sleep or Deep Sleep, it gets into low power consumption mode.

**Table 3-3 XRAM Write Protection Register**

| 0x1B20, R/W | Write Protection Register of upper 128 bytes of XRAM; only bit0 can be read out, XRAMPwd | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Function | | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| Disable Write Protection | 0x33 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| Restore Write Protection | 0x32 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| Default Value | 0x00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# 3.3. Program Memory Space

The addresses of 64-Kbyte FLASH, FLASH Control Register and program encryption bytes are mapped into the program memory space (Figure 3-3).

The FLASH Control Registers (0xFFFE and 0xFFFF) can control the programming mode and power consumption mode of the FLASH. Writing 0x86 into FLASH Control Register (0xFFFE) can reduce power consumption of FLASH without any compromise of performance.

```
unsigned char xdata *p;

SPC_FNC = 0x01;

p = 0xFFFE;

*p = 0x86;

SPC_FNC = 0x00;
```

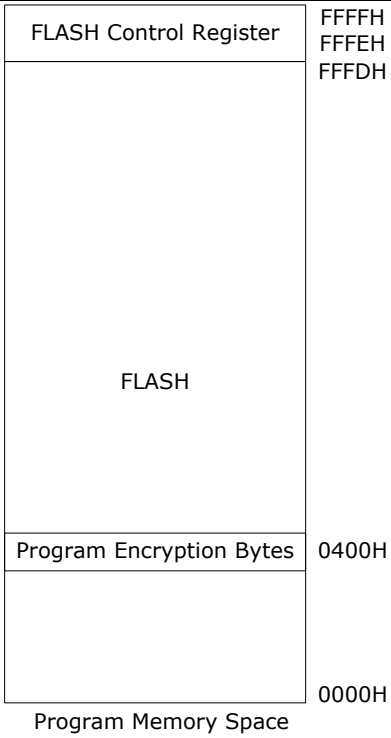



**Figure 3-3 Program Memory Space**

# 3.4. Debugging, ISP and FLASH Encryption

## 3.4.1. Debugging

When the level on the pin MODE0 is low, the chip is in the debugging mode. When debugging, POR/BOR and the WDT reset are masked, the WDT reset cannot configure the Wake-up/Reset Reason Query Register (WRS) to 0x01, and the P0 port is configured as JTAG interface via which the real-time debugging circuits of internal hardware can be accessed for debugging and ISP programming. No capacitor should be connected to P0 ports to enable downloading Hex File into the chip.

## 3.4.2. ISP

Use the specific DLL and the Keil µVision IDE to do debug and ISP operations. When debugging is in process, please comment the following lines out of the application software, or abnormal debug conditions may occur.

- Switching the system clock from PLL output to OSC output
- Getting into Sleep, or Deep Sleep

### 3.4.3. FLASH Encryption

Bit 0 in the byte 0400h of FLASH memory is the encryption bit.

The encryption bit has no effect on the mass erasing operation and writing operation on the range 0000h~FFFFh.

When the encryption bit is cleared, the reading operation on the space ranged from 0000h~BFFFh, and the page-erasing operation on the space ranged from 0000h~FFFFh, is prohibited, which will be read out as 0. But it has no effect on the reading operation on the space ranged from C000h~FFFFh.

When the encryption bit is set to 1, the space ranged from 0000h ~ FFFFh can be accessed anyway.

Table 3-4 lists the access prohibition of ISP operation to FLASH memory.

**Table 3-4 Access Prohibition of ISP Operation to Flash Space**

| Access | Address | Encryption bit is cleared | | | | Encryption bit is set to 1 | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Mass erase | Write | Read | Page erase | Mass erase | Write | Read | Page erase |
| ISP (through JTAG) | 0000h~BFFFh | √ | √ | X | X | √ | √ | √ | √ |
| | C000h~FFFFh | √ | √ | √ | X | √ | √ | √ | √ |

The POR or RST Pin reset is needed to enable the encryption after ISP operation.

## 3.5. CPU

### 3.5.1. Instruction Set

The instruction set of CPU is compatible with the industry standard 8051 in binary code and the execution results are functionally equivalent. However, the number of clock cycles that each instruction cycle needs is different from standard 8051 instruction set. And the execution timing of each instruction is also different from that of standard 8051. Each CPU instruction cycle includes four clock cycles.

**Table 3-5 Instruction Set**

| Symbol | Description | | | |
|---|---|---|---|---|
| A | Accumulator | | | |
| Rn | Register R0-R7 | | | |
| direct | Internal direct addressable registers | | | |
| @Ri | Internal register pointed to by R0 or R1 (except MOVX) | | | |
| rel | Two's complement offset byte | | | |
| bit | Direct bit address | | | |
| #data | 8-bit constant | | | |
| #data 16 | 16-bit constant | | | |
| addr 16 | 16-bit destination address | | | |
| addr 11 | 11-bit destination address | | | |
| Mnemonic | Description | Byte | Inst. Cycles | Hex Code |
| Arithmetic | | | | |
| ADD A, Rn | Add register to A | 1 | 1 | 28 – 2F |
| ADD A, direct | Add direct byte to A | 2 | 2 | 25 |
| ADD A, @Ri | Add data memory to A | 1 | 1 | 26 – 27 |
| ADD A, #data | Add immediate to A | 2 | 2 | 24 |
| ADDC A, Rn | Add register to A with carry | 1 | 1 | 38 – 3F |
| ADDC A, direct | Add direct byte to A with carry | 2 | 2 | 35 |
| ADDC A, @Ri | Add data memory to A with carry | 1 | 1 | 36 – 37 |
| ADDC A, #data | Add immediate to A with carry | 2 | 2 | 34 |
| SUBB A, Rn | Subtract register from A with borrow | 1 | 1 | 98 – 9F |
| SUBB A, direct | Subtract direct byte from A with borrow | 2 | 2 | 95 |
| SUBB A, @Ri | Subtract data memory from A with borrow | 1 | 1 | 96 – 97 |
| SUBB A, #data | Subtract immediate from A with borrow | 2 | 2 | 94 |
| INC A | Increment A | 1 | 1 | 04 |
| INC Rn | Increment register | 1 | 1 | 08 – 0F |

| | | | | |
|---|---|---|---|---|
| INC direct | Increment direct byte | 2 | 2 | 05 |
| INC @Ri | Increment data memory | 1 | 1 | 06 – 07 |
| DEC A | Decrement A | 1 | 1 | 14 |
| DEC Rn | Decrement register | 1 | 1 | 18 – 1F |
| DEC direct | Decrement direct byte | 2 | 2 | 15 |
| DEC @Ri | Decrement data memory | 1 | 1 | 16 – 17 |
| INC DPTR | Increment data pointer | 1 | 3 | A3 |
| MUL AB | Multiply A by B | 1 | 5 | A4 |
| DIV AB | Divide A by B | 1 | 5 | 84 |
| DA A | Decimal adjust A | 1 | 1 | D4 |
| Logical | | | | |
| ANL A, Rn | AND register to A | 1 | 1 | 58 – 5F |
| ANL A, direct | AND direct byte to A | 2 | 2 | 35 |
| ANL A, @Ri | AND data memory to A | 1 | 1 | 56 – 57 |
| ANL A, #data | AND immediate to A | 2 | 2 | 54 |
| ANL direct, A | AND A to direct byte | 2 | 2 | 52 |
| ANL direct, #data | AND immediate data to direct byte | 3 | 3 | 53 |
| ORL A, Rn | OR register to A | 1 | 1 | 48 – 4F |
| ORL A, direct | OR direct byte to A | 2 | 2 | 45 |
| ORL A, @Ri | OR data memory to A | 1 | 1 | 46 – 47 |
| ORL A, #data | OR immediate to A | 2 | 2 | 44 |
| ORL direct, A | OR A to direct byte | 2 | 2 | 42 |
| ORL direct, #data | OR immediate data to direct byte | 3 | 3 | 43 |
| XRL A, Rn | Exclusive-OR register to A | 1 | 1 | 68 – 6F |
| XRL A, direct | Exclusive-OR direct byte to A | 2 | 2 | 65 |
| XRL A, @Ri | Exclusive-OR data memory to A | 1 | 1 | 66 – 67 |
| XRL A, #data | Exclusive-OR immediate to A | 2 | 2 | 64 |
| XRL direct, A | Exclusive-OR A to direct byte | 2 | 2 | 62 |

| | | | | |
|---|---|---|---|---|
| XRL direct, #data | Exclusive-OR immediate to direct byte | 3 | 3 | 63 |
| CLR A | Clear A | 1 | 1 | E4 |
| CPL A | Complement A | 1 | 1 | F4 |
| SWAP A | Swap nibbles of A | 1 | 1 | C4 |
| RL A | Rotate A left | 1 | 1 | 23 |
| RLC A | Rotate A left through carry | 1 | 1 | 33 |
| RR A | Rotate A right | 1 | 1 | 03 |
| RRC A | Rotate A right through carry | 1 | 1 | 13 |
| Data Transfer | | | | |
| MOV A, Rn | Move register to A | 1 | 1 | E8 – EF |
| MOV A, direct | Move direct byte to A | 2 | 2 | E5 |
| MOV A, @Ri | Move data memory to A | 1 | 1 | E6 – E7 |
| MOV A, #data | Move immediate to A | 2 | 2 | 74 |
| MOV Rn, A | Move A to register | 1 | 1 | F8 – FF |
| MOV Rn, direct | Move direct byte to register | 2 | 2 | A8 – AF |
| MOV Rn, #data | Move immediate to register | 2 | 2 | 78 – 7F |
| MOV direct, A | Move A to direct byte | 2 | 2 | F5 |
| MOV direct, Rn | Move register to direct byte | 2 | 2 | 88 – 8F |
| MOV direct, direct | Move direct byte to direct byte | 3 | 3 | 85 |
| MOV direct, @Ri | Move data memory to direct byte | 2 | 2 | 86 – 87 |
| MOV direct, #data | Move immediate to direct byte | 3 | 3 | 75 |
| MOV @Ri, A | MOV A to data memory | 1 | 1 | F6 – F7 |
| MOV @Ri, direct | Move direct byte to data memory | 2 | 2 | A6 – A7 |
| MOV @Ri, #data | Move immediate to data memory | 2 | 2 | 76 – 77 |
| MOV DPTR, #data | Move immediate to data pointer | 3 | 3 | 90 |
| MOVC A, @A+DPTR | Move code byte relative DPTR to A | 1 | 3 | 93 |
| MOVC A, @A+PC | Move code byte relative PC to A | 1 | 3 | 83 |
| MOVX A, @Ri | Move external data (A8) to A | 1 | 2 – 9* | E2 – E3 |

| | | | | |
|---|---|---|---|---|
| MOVX A, @DPTR | Move external data (A16) to A | 1 | 2 – 9* | E0 |
| MOVX @Ri, A | Move A to external data (A8) | 1 | 2 – 9* | F2 – F3 |
| MOVX @DPTR, A | Move A to external data (A16) | 1 | 2 – 9* | F0 |
| PUSH direct | Push direct byte onto stack | 2 | 2 | C0 |
| POP direct | Pop direct byte from stack | 2 | 2 | D0 |
| XCH A, Rn | Exchange A and register | 1 | 1 | C8 – CF |
| XCH A, direct | Exchange A and direct byte | 2 | 2 | C5 |
| XCH A, @Ri | Exchange A and data memory | 1 | 1 | C6 – C7 |
| XCHD A, @Ri | Exchange A and data memory nibble | 1 | 1 | D6 – D7 |
| Boolean | | | | |
| CLR C | Clear carry | 1 | 1 | C3 |
| CLR bit | Clear direct bit | 2 | 2 | C2 |
| SETB C | Set carry | 1 | 1 | D3 |
| SETB bit | Set direct bit | 2 | 2 | D2 |
| CPL C | Complement carry | 1 | 1 | B3 |
| CPL bit | Complement direct bit | 2 | 2 | B2 |
| ANL C, bit | AND direct bit to carry | 2 | 2 | 82 |
| ANL C, /bit | AND direct bit inverse to carry | 2 | 2 | B0 |
| ORL C, bit | OR direct bit to carry | 2 | 2 | 72 |
| ORL C, /bit | OR direct bit inverse to carry | 2 | 2 | A0 |
| MOV C, bit | Move direct bit to carry | 2 | 2 | A2 |
| MOV bit, C | Move carry to direct bit | 2 | 2 | 92 |
| Branching | | | | |
| ACALL addr 11 | Absolute call to subroutine | 2 | 3 | 11 – F1 |
| LCALL addr 16 | Long call to subroutine | 3 | 4 | 12 |
| RET | Return from subroutine | 1 | 4 | 22 |
| RETI | Return from interrupt | 1 | 4 | 32 |
| AJMP addr 11 | Absolute jump unconditional | 2 | 3 | 01 – E1 |

| | | | | |
|---|---|---|---|---|
| LJMP addr 16 | Long jump unconditional | 3 | 4 | 02 |
| SJMP rel | Short jump (relative address) | 2 | 3 | 80 |
| JC rel | Jump on carry = 1 | 2 | 3 | 40 |
| JNC rel | Jump on carry = 0 | 2 | 3 | 50 |
| JB bit, rel | Jump on direct bit = 1 | 3 | 4 | 20 |
| JNB bit, rel | Jump on direct bit = 0 | 3 | 4 | 30 |
| JBC bit, rel | Jump on direct bit = 1 and clear | 3 | 4 | 10 |
| JMP @A + DPTR | Jump indirect relative DPTR | 1 | 3 | 73 |
| JZ rel | Jump on accumulator = 0 | 2 | 3 | 60 |
| JNZ rel | Jump on accumulator ≠ 0 | 2 | 3 | 70 |
| CJNE A, direct, rel | Compare A, direct JNE relative | 3 | 4 | B5 |
| CJNE A, #d, rel | Compare A, immediate JNE relative | 3 | 4 | B4 |
| CJNE Rn, #d, rel | Compare reg, immediate JNE relative | 3 | 4 | B8 – BF |
| CJNE @Ri, #d, rel | Compare ind, immediate JNE relative | 3 | 4 | B6 – B7 |
| DJNZ Rn, rel | Decrement register, JNZ relative | 2 | 3 | D8 – DF |
| DJNZ direct, rel | Decrement direct byte, JNZ relative | 3 | 4 | D5 |
| Miscellaneous | | | | |
| NOP | No operation | 1 | 1 | 00 |
| There is an additional reserved opcode (A5) that performs the same function as NOP. | | | | |

* Number of cycles is user-selectable.

## 3.5.2. Programmable MOVX Timing

The programmable MOVX timing feature enables application software to adjust the speed of data memory or registers access. CPU can execute the MOVX instruction in as little as two instruction cycles. However, it is sometimes desirable to stretch this value. The three lower bits (MD2~0, CKCON.2~0) of the register CKCON SFR (0x8E) control the stretch value. These three bits can set the stretch values from 0 to 7. The stretch value of 0 adds zero instruction cycle, resulting in MOVX instructions executing in two instruction cycles. The stretch value of 7 adds seven instruction cycles, resulting in MOVX instructions executing in nine instruction cycles. The stretch values can be changed dynamically under program control. The stretch values affect the width of

the read/write strobe and all related timing. A higher stretch value results in a wider read/write strobe. By default, the stretch value is 1 (three cycle MOVX).

**Table 3-6 Programmable MOVX Timing**

| CKCON.2 | CKCON.1 | CKCON.0 | Memory Cycles | Read/Write Strobe Width (clocks) |
|---|---|---|---|---|
| MD2 | MD1 | MD0 | | |
| 0 | 0 | 0 | 2 | 2 |
| 0 | 0 | 1 | 3 (default value) | 4 |
| 0 | 1 | 0 | 4 | 8 |
| 0 | 1 | 1 | 5 | 12 |
| 1 | 0 | 0 | 6 | 16 |
| 1 | 0 | 1 | 7 | 20 |
| 1 | 1 | 0 | 8 | 24 |
| 1 | 1 | 1 | 9 | 28 |

## 3.5.3. Dual Data Pointers

Dual data pointers can improve the efficiency significantly when moving large blocks of data.

The standard data pointer in CPU is DPTR0 at SFR locations 0x82 and 0x83. The second data pointer (DPTR1) at SFR locations 0x84 and 0x85 is added. The bit SEL (bit0) in the DPTR Select Register (DPS SFR, 0x86), selects the active pointer. When SEL = 0, DPL0 and DPH0 are used. When SEL = 1, DPL1 and DPH1 are used. No other bits of DPS SFR are used.

All DPTR-related instructions use the currently selected data pointer. Configure the SEL bit to switch the active pointer. The fastest way to do so is to use the increment instruction (INC DPS). This requires only one instruction to switch from a source address to a destination address, and storing the source and destination addresses is not needed when doing a block move, which can save the number of application codes.

# 4. Reset

The reset circuits can be categorized into 3 levels.

- Level 3, including WDT Reset and Debug Reset, can reset CPU and peripheral circuits, energy metering access registers, analog control registers (0x1B00~0x1B17), system clock control registers, and FLASH control registers;
- Level 2, including IO Wake-up Reset, RTC Wake-up Reset and Recovery Reset, besides those reset by Level 3 resets, also can reset WDT;
- Level 1, including RST Pin Reset and POR/BOR, besides those reset by Level 2 resets, also can reset LCD and system status control registers.

Any reset has no effect on the data stored in RTC timing registers, IRAM and XRAM. When the level on the pin MODE0 is low, POR/BOR and WDT Reset are masked, and the register Wake-up/Reset Reason Query Register (WRS) cannot be configured to 0x01.

Figure 4-1 displays the reset circuit architecture.

## 4.1. Level 3

Level 3 resets include WDT reset and Debug reset. They can reset CPU, interrupts, timers, UARTs, IOs, energy metering access registers, analog control registers (0x1B00~0x1B17), system clock control registers, and FLASH control registers.

### 4.1.1. WDT Reset

A reset is generated if the WDT is not cleared in 1.5 seconds. If the WDT is not cleared in 2 seconds, a WDT reset is generated again. The WDT reset goes on 20 ms.

In the Debug mode (the level on the pin MODE0 is low), the WDT Reset is masked, and the register Wake-up/Reset Reason Query Register (WRS) cannot be configured to 0x01.

### 4.1.2. Debug Reset

When reset debugging is asked, the Debug Reset occurs. When debugging in the IDE, click "reset" to call for reset debugging, and the Debug Reset occurs. When programming with ISP programmer, the programmer gives the debug reset instruction.

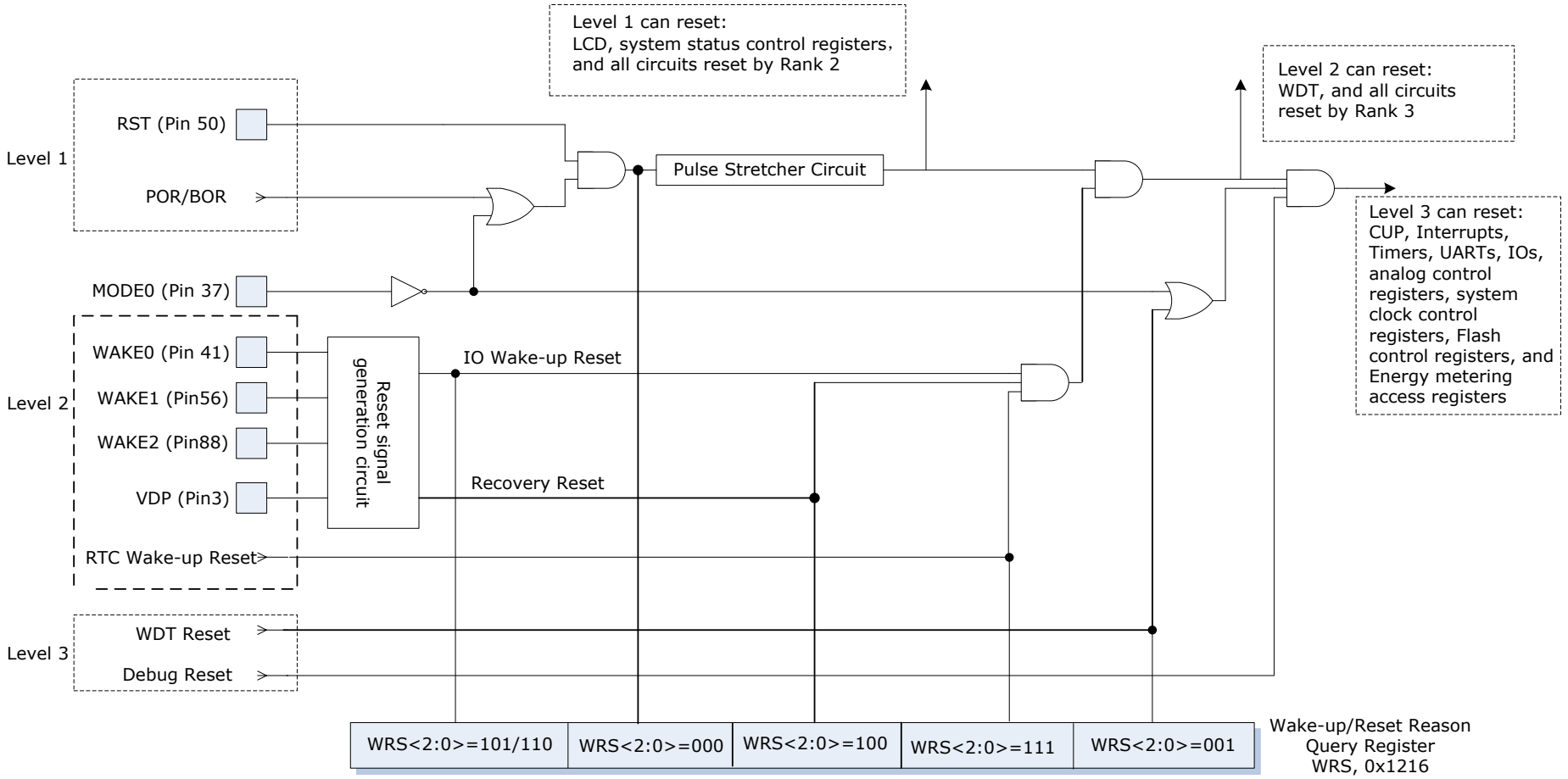



**Figure 4-1 Reset Circuit**

# 4.2. Level 2

Level 2 resets include IO Wake-up Reset, RTC Wake-up Reset and Recovery Reset. They can reset all on-chip circuits except RTC timing registers, LCD and system status control registers.

## 4.2.1. Recovery Reset

When the input level on the pin VDP is from lower than 1.0V to higher than 1.1V, or, the input level on the pin VDP is higher than 1.1V after Level 1 resets, the recovery reset occurs. It goes on 8 OSC clock periods.

## 4.2.2. IO Wake-up Reset

Configure the pins WAKE0 (P1.0) and WAKE1 (1.2) as input. In *Sleep mode*, when any one port of the both jumps from high level to low level, and they go on 4 clock periods or more in both levels, the IO Wake-up Reset happens.

Or, configure the pin WAKE2 (2.6) as input. In *Sleep or Deep Sleep mode* ("System Status" gives the definition of the modes Sleep and Deep Sleep), when this port jumps from high level to low level, and it goes on 4 clock periods or more in both levels, the IO Wake-up Reset happens.

## 4.2.3. RTC Wake-up Reset

In Sleep mode, the RTC outputs reset signals periodically according to the given wake-up interval.

# 4.3. Level 1

Level 1 includes RST Pin Reset and POR/BOR. They can reset all on-chip circuits, except the RTC timing registers.

## 4.3.1. RST Pin Reset

The reset input signal on the pin RST is extended by internal circuit as illustrated in Figure 4-2 and Figure 4-3.

If the width of RST input signal is less than an OSC clock period, this reset signal is extended by internal circuit to four OSC clock periods, as shown in Figure 4-2.

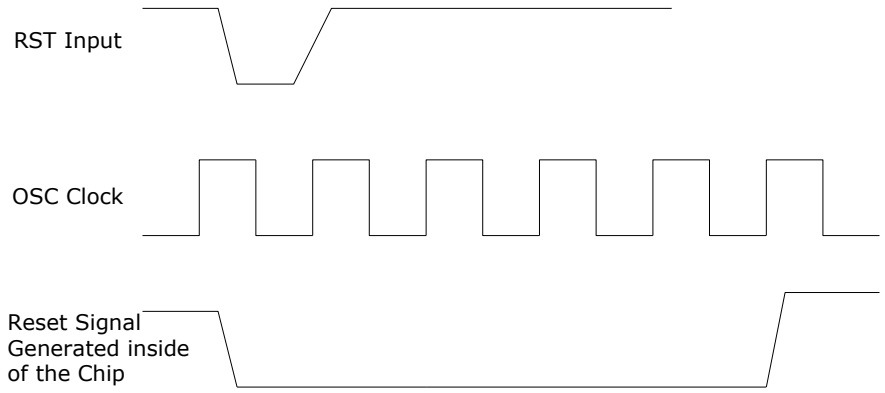



**Figure 4-2 RST Input is Extended**

If the width of RST input signal is more than an OSC clock period, another four OSC clock periods are added to this reset signal by internal circuit of the SoC (Figure 4-3).

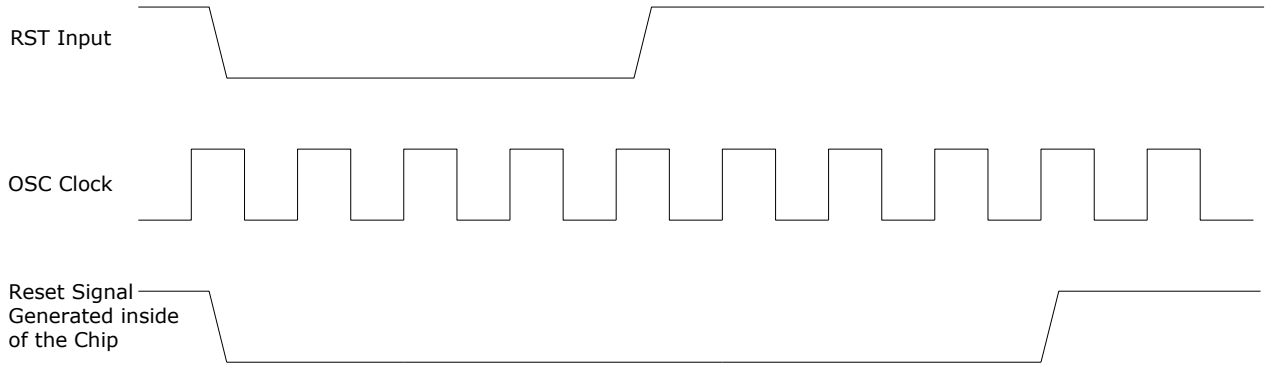



**Figure 4-3 RST Input is Extended**

## 4.3.2. POR/ BOR

The POR circuit is used to monitor the level of the LDO25 output. When the output level is higher than 1.8V, the reset signal is released. When it is lower than 1.8V, the reset output signal is active. The threshold value, here the 1.8V, depends on the chip type. In power-down, the BOR circuit outputs reset signal when the LDO25 level is lower than 1.7V. POR/BOR input also can be extended, as shown in Figure 4-3. The threshold level is independent of the configuration of LDO25 voltage.

When debugging (the level on the pin MODE0 is low), POR/BOR are masked.

# 4.4. Reset Reason

When a reset occurs, the reset reason can be detected via querying the register in Table 4-1.

**Table 4-1 Wake-up/Reset Reason Query Register (WRS, 0x1216)**

| 0x1216, R/W | Wake-up/Reset Reason Query Register, WRS | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | | | | | | WRS<2> | WRS<1> | WRS<0> |

**Table 4-2 Configuring WRS**

| Configuration | Description |
|---|---|
| 0x00 | RST Pin Reset, or POR/BOR. To wake up the system from Sleep and Deep Sleep mode. |
| 0x01 | WDT Reset. Unable to wake up the system from Sleep or Deep Sleep mode. |
| 0x04 | Recovery Reset. To wake up the system from Sleep and Deep Sleep mode. |
| 0x05 | IO Wake-up Reset. Only to wake up the system from Sleep mode. Active on falling edges of WAKE0 and WAKE1. |
| 0x06 | IO Wake-up Reset. To wake up the system from Sleep and Deep Sleep mode. Active on the falling edge of WAKE2. |
| 0x07 | RTC Wake-up Reset. To wake up the system from Sleep mode. |

# 5. Clock

## 5.1. Features

The clock system includes OSC crystal oscillator circuit, RC and PLL. The off-chip 32768Hz and the on-chip oscillator generate 32768Hz OSC clock. The PLL locks on a multiple of this frequency to provide a stable clock for the MCU and ADC.

The clock circuits of V9003/V9103 are of power-saving design and the clock frequency is flexibly configurable.

**Table 5-1 System Clock Sources**

| Clock | Frequency | Source | Description |
|---|---|---|---|
| OSC | 32.768kHz | Crystal Oscillator | It takes tens of milliseconds to start when being powered up. |
| PLL | 6.5536MHz | PLL | The frequency is configurable. The highest frequency is 13.1072MHz. When PLL is turned off, the frequency is switched to 32.768kHz automatically. |
| RC | 32.768kHz | RC Oscillator | It works when powering up rapidly. No configuration is needed. The frequency depends on the chip type and the temperature. |

## 5.2. OSC Clock Generating Circuit

The OSC clock generating circuit works all the time. It is a low power-consuming block. After reset, this circuit consumes 1μA which can be lowered to 0.5μA via the MCU configuration. The resistor and capacitor in the embedded oscillator circuit of V9003/V9103 are configurable, and the OSC output frequency can be adjusted finely via this configuration.

Configure the bit XEASYN in the Clock Configuration Register (CLKCFG, Table 5-2) to select the crystal frequency and power-consumption. By default, high power-consumption and crystal frequency are selected.

**Table 5-2 Clock Configuration Register (CLKCFG, 0x1B13)**

| 0x1B13, R/W | Clock Configuration Register, CLKCFG | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | | | CLKOSEL* | ADCLKEXT* | XEASYN | XTRIM<2> | XTRIM<1> | XTRIM<0> |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

*The bits CLKOSEL and ADCLKEXT of the register CLKCFG have nothing to do with the operation of OSC.

**Table 5-3 Bit Description of CLKCFG**

| Bit | Description | Remark |
|---|---|---|
| XTRIM<2:0> | To control the crystal frequency via adjusting the capacitors in the oscillator circuit | 000~111, 1pF~7pF |
| XEASYN | Power-consumption mode of the crystal | 0, high power-consumption (1μA);<br>1, low power-consumption (0.5μA). |
| ADCLKEXT | To select the clock source for ADC | 0, PLL clock; 1, 1/4 OSC clock |
| CLKOSEL | To select the ADC clock output | 0, in-phase of input clock; 1, out-of-phase of input clock |

The ambient temperature has important effect on the OSC output frequency. So, the RTC should be calibrated when high accuracy is required.

# 5.3. PLL Generating Circuit

The registers related to PLL configuration are as follows.

**Table 5-4 PLL Control Register (PLLCtrl, 0x1B12)**

| 0x1B12, R/W | PLL Control Register, PLLCtrl | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | | | PLLPDN | BGPPDN | | | | |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 5-5 Bit Description of PLLCtrl**

| Bit | Description | Remark |
|---|---|---|
| PLLPDN | To turn on or turn off PLL. | 0, Off; 1, On. |
| BGPPND | To turn on or turn off BandGap. | 0, Off; 1, On. Turn on BandGap, and then turn on PLL. |

**Table 5-6 Frequency Configuration Register 1 (FRQCFG0, 0x1B15)**

| 0x1B15, R/W | Frequency Configuration Register 1, FRQCFG0 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | MEAS <2> | MEAS <1> | MEAS <0> | PLLSEL | ADCLKSEL <1> | ADCLKSEL <0> | MCUCLKSEL <1> | MCUCLKSEL <0> |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 5-7 Bit Description of FRQCFG0**

| Bit | Description | Remark |
|---|---|---|
| MCUCLKSEL<1:0> | To select the clock frequency of MCU | 3.2768MHz<br>00, ×1; 01, ×2; 10, ×4; 11, × 0.5 |
| ADCLKSEL<1:0> | To select the clock frequency of ADC | 409.6 kHz<br>00, ×1; 01, ×2; 10, ×4; 11, × 0.5 |
| PLLSEL | To select the grid power, 50Hz or 60Hz | 0, 50Hz; 1, 60Hz |

**Table 5-8 Frequency Configuration Register 2 (FRQCFG1, 0x1B16)**

| 0x1B16, R/W | Frequency Configuration Register 2, FRQCFG1 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | | MEACLKSEL <1> | MEACLKSEL <0> | WDCKSEL | | | PLLIB <1> | PLLIB <0> |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 5-9 Bit Description of FRQCFG1**

| Bit | Description | Remark |
|---|---|---|
| PLLIB<1:0> | To configure the charge pump current of PLL | 00, ×1; 01, ×2; 10, ×2; 11, ×3 |
| MEACLKSEL<1:0> | To select the clock frequency of the energy metering DSP | 3.2768MHz<br>00, ×1; 01, ×2; 10, ×4; 11, ×0.5 |
| WDCKSEL | To select the clock source of WDT | 0, RC clock; 1, OSC clock |

**Note:** The frequency for ADC should be one eighth of that for the energy metering DSP, which means that the value for MEACLKSEL [1:0] should be equal to that for ADCLKSEL [1:0].

**Table 5-10 Analog Signal Control Register (AFSIG, 0x1B50)**

| 0x1B50, R/W | Analog Signal Control Register, AFSIG | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | Bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | PLLLCK | - | PWRUP | PWRDN | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- PLLLCK: 1, PLL is locked; 0, PLL is not locked. Besides this flag, PLL lock interrupt also can be used to detect the status of PLL.
- PWRUP: to detect the input voltage on the pin VDP. When the level is higher than 1.1V, PWRUP = 1; when the level is lower than 1.0V, PWRUP = 0. PWRUP=1, main power supply, or switching from battery power supply to main power supply. 0, battery power supply, or switching from main power supply to battery power supply.
- PWRDN: to detect the input level on the pin VDP. When the voltage is lower than 1.0V, PWRDN = 1; When the level is higher than 1.1V, PWRDN = 0. PWRDN=1, power down. The indicating function of PWRDN is on. 0, no power down. If the power down interrupt is enabled, the power-down event can generate an interrupt to CPU.

In the 50Hz power grid, the PLL provides 32768Hz OSC clock until the MCU is turned on. The PLL generates 3 clocks for the MCU, the energy metering DSP and the ADC, respectively. For MCU, the frequency can be configured to 32.768kHz, 3.2768MHz, 6.5536MHz, 1.6384MHz and 13.1072MHz; for the energy metering DSP, the frequency can be configured to 3.2768MHz, 6.5536MHz, 13.1072MHz and 1.6384MHz; For ADC, the frequency can be configured to 204.8kHz, 409.6kHz, 819.2kHz, 1638.4kHz, and 3276.8kHz. The frequency for the energy metering DSP should be 8 times of that for ADC. When working normally, 6.5536MHz is used as the frequency for the energy metering DSP, 819.2kHz is used for the ADC, and the power consumption of the whole chip is about 13mA. In the low power-consumption mode, the PLL output frequency is lowered and the power-consumption of the ADC is also lowered to save power for the whole

system. In the normal working mode, the minimum frequency for MCU is 32.768kHz, for the energy metering DSP is 1.6384MHz, and the synchronous frequency of ADC is 204.8kHz.

The PLL also can be configured to work in the 60Hz power grid, in which the PLL provides frequency at 1.2 times higher than that generated in 50Hz power grid.

PLL should be turned on as follows:

1. Use PLL Control Register (PLLCtrl) to turn on or off the PLL and configure it.
2. Wait for PLL frequency configuration and being locked. There are two methods to detect PLL lock: querying the flag PLLLCK in the Analog Signal Control Register (AFSIG) or the PLL lock interrupt (Interrupt Resources).
3. After the PLL is locked, follow this method to use the PLL as the clock source: configure the flag CLKS in the register CLKS (0x1214) to be 1.

**Note:**

1. Use the following steps to switch the frequency of PLL:
   1.1 Switch to the 32.768kHz OSC frequency as the MCU frequency;
   1.2 Turn off PLL;
   1.3 Turn on PLL;
   1.4 Adjust the clock frequency;
   1.5 Wait for the PLL being locked;
   1.6 Use the PLL clock as the clock source.
2. Switch the system clock source into the OSC clock to reconfigure PLL when PLL is on.

The V9003/V9103 can work in 50Hz and 60Hz power grids. It can be configured as follows: when the bit PLLSEL in the register FRQCFG0 (0x1B15) is set to 0, 50Hz power grid is selected; when it is set to 1, 60Hz power grid is selected. By default, it is configured for 50Hz power grid. When it is configured for 60Hz power grid, the output frequency of PLL should be 1.2 times higher than that in 50Hz power grid, so some clock frequency related configuration, such as the baud rate and timers, should be reconfigured.

# 5.4. Clock Control Circuit

According to the output frequency, the clocks in the clock control circuit can be categorized into:

- Clock 1: for CPU, RAM, FLASH, extended interrupts, extended timers/UARTs and IOs;

- Clock 2: for energy metering circuits;
- Clock 3: for LCD;
- Clock 4: for WDT;
- Clock 5: for RTC.

The frequencies and on-off of all the above clocks are controlled by the clock control circuit. Clock 1 and Clock 2 can use both OSC and PLL clock as the clock source. Clock 4 uses RC oscillator clock as the clock source. Clock1, Clock 2, Clock 3 and Clock 4 all can be turned off, and after turning off them, the corresponding functional blocks using these clock output frequencies cannot work.

**Table 5-11 Clocks in Clock Control Circuit**

| | Clock 1 | Clock 2 | Clock 3 | Clock 4* | Clock 5 |
|---|---|---|---|---|---|
| Status After Reset | OSC, on | OSC, on | OSC, on | RC, on | OSC, on |
| From OSC to PLL | PLL | PLL | OSC | RC | OSC |
| From PLL to OSC | OSC | OSC | OSC | RC | OSC |
| Turn off | SYSG (Bit0, 0x1213) | PMG = 1 (Bit0, 0x1211) | LCDG = 0 (Bit 0, 0x1210) | SYSG (0x1213) | Unable to be turned off. |

*Clock 4 source can be switched into OSC clock from RC clock. However, to ensure the stabilization of WDT, this switchover is not recommended.

# 5.5. Normal Mode for Switchover between PLL and OSC

## 5.5.1. Related Registers to System Clock Control

Table 5-12 lists all related registers to system clock control.

**Table 5-12 Related Registers to System Clock Control**

| Address | Register | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| 0x1210 | LCDG | 0 | 1 | 0 | 1 | 1 | 0 | 1 | LCDG |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0x1211 | PMG | 0 | 1 | 0 | 1 | 1 | 0 | 1 | PMG |
| 0x1213 | SYSG | 0 | 1 | 0 | 1 | 1 | 0 | 1 | SYSG |
| 0x1214 | CLKS | 0 | 1 | 0 | 1 | 1 | 0 | 1 | CLKS |
| 0x1215 | DEEP | 0 | 1 | 0 | 1 | 1 | 0 | 1 | DEEP |

**Table 5-13 Bit Description of System Clock Control Related Registers**

| Bit | Function | Description | Reset Status |
|---|---|---|---|
| LCDG | To turn off the clock for LCD. | 0, On; 1, Off. | When POR occurs, LCDG= 1;<br>When other resets occur, LCDG=0. |
| PMG | To turn off clocks for the energy metering DSP and ADC. | 0, On; 1, Off | 1 |
| SYSG | To turn off Clock 1, and get into Sleep mode. | 0, On; 1, Off | 0 |
| CLKS | To select the clock source for Clock 1. | 0, OSC; 1, PLL | 0 |
| DEEP | To select the Sleep mode. | 0, Sleep; 1, Deep Sleep | 0 |

To protect the system from the error, the write operation of the above registers should follow the process as:

1. Write the access password 0xD6 into the register located at the address 0x1217;
2. Write the access password 0xAD into the register located at the address 0x1218 to enable accessing to the clock control registers (0x1210~0x1215). Both write operations should be continuous;
3. Write the password 0101101 into Bit7~Bit1 of the clock control registers (0x1210~0x1215), and then, write the appropriate value into the Bit0 of them.
4. Configure the system status via configuring the system clock control registers;
5. After configuration, write 0xAE into the control register located at the address 0x1218 to protect the clock control registers (0x1210~0x1215) from access.

Read of the register is not protected. Bit7~Bit1 should be read out as 0.

## 5.5.2. Status after Reset

After the Level 1 or Level 2 resets, all bits in analog control registers (0x1B00~0x1B17) are configured to 0, PLL is off, Clock 1 is on (0x1214, CLKS=0), and use OSC as the clock source (0x1213, SYSG=0).

## 5.5.3. Switchover from OSC to PLL

Access the register PLLCtrl (0x1B12), FRQCFG0 (0x1B15) and FRQCFG1 (0x1B16) to turn on PLL and configure the output frequency to be, for example, 3.2768MHz.

Write 1 into the flag CLKS (in the register CLKS, 0x1214), and the clock source of Clock 1 is switched from OSC to PLL, which needs only one PLL clock period. When PLL is off, writing 1 into the flag CLKS means that the switchover from OSC to PLL is not available. If PLL is turned off abnormally when it is used as the clock source, the output frequency will turn into 32768Hz automatically, but the flag CLKS is still 1. So, the current clock status can be detected via the following sources:

- PLLLOCK interrupt. This is the PLL lock signal on high level, indicating PLL is stable, which is synchronized before input into the chip. This interrupt can be generated on rising or falling edges. On the falling edge of the signal, PLL is turned off.
- The flag PLLLCK in the register AFSIG (bit4, 0x1450).

## 5.5.4. Switchover from PLL to OSC

Write 0 into the flag CLKS (in the register CLKS, 0x1214) to switch the clock source of Clock 1 from PLL to OSC. From writing 0 into the flag CLKS (in the register CLKS, 0x1214) to finishing the switchover, it needs only less than one OSC clock period. In this period, write operation to all analog control registers (0x1B00~0x1B17) is not available. Therefore, after writing 0 into the flag CLKS (in the register CLKS, 0x1214), read out this flag repeatedly till its value changes from 1 to 0, which means switching operation is finished. Then further tasks can be executed.

## 5.5.5. Turning off Clock

When PWRUP=0 (in the register AFSIG), and CLKS=0 (in the register CLKS, 0x1214), write 1 into the flag SYSG (in the register SYSG, 0x1213) to turn off Clock 1. When Clock 1 is turned off, the circuits using Clock 1 stop working.

Set the flags LCDG (in the register LCDG) and PMG (in the register PMG) to 1 to turn off the clock for LCD and the energy metering modules any time.

# 5.6. Quick Mode for Switchover between PLL and OSC

Access to Bit5 and Bit1 in the register PSW SFR (0xD0) to complete the switchover between PLL and OSC follows the quick mode.

**Table 5-14 Program Status Word SFR**

| 0xD0, R/W | PSW SFR | |
|---|---|---|
| | Bit5 | Bit1 |
| | F0 | F1 |
| Default Value | 0 | 0 |

In Table 5-14,

- F0: Write 1 into F0 to switch the clock sources of Clock 1 and Clock 2 from PLL to OSC output frequency, to trigger the hardware to turn off PLL automatically, and to turrn off Clock 1 (for CPU and peripherals) and Clock 4 (for WDT). The system goes into sleep mode.
- F1: Write 1 into F1 to trigger the hardware to turn on PLL automatically, to switch the clock frequency of Clock 1 and Clock 2 from OSC to PLL. When PLL is turned on by the hardware, the output frequency is 6.5536 MHz.

**Table 5-15 PLL and System Clock Status**

| F1, F0 | PLL and System Clock Status |
|---|---|
| 0, 0 | The system operation depends on the program, and the program decides whether to trigger the hardware to turn on PLL and the switchover between PLL and OSC or not. |
| 0, 1 | From PLL to OSC for Clock 1 and Clock 2; PLL, Clock 1 (for CPU and peripherals), and Clock 4 (for WDT), off. |
| 1, 0 | PLL, on; Output frequency, 6.5536MHz; From OSC to OLL for Clock 1 and Clock 2. |
| 1, 1 | From PLL to OSC for Clock 1 and Clock 2; PLL, Clock 1 (for CPU and peripherals), and Clock 4 (WDT), off. |

When compiling C programs via the KEIL compiler, if the signed integer division and the functions in math.h are used, 1 is automatically written into the bit F0, and the clock source switchover between Clock 1 and Clock 2 is abnormal.

### 5.6.1. Status after Reset

After Level 1 and Level 2 resets, F0 and F1 are configured to be 00, when, the system operation depends on the program, and the program decides whether to trigger the hardware to turn on PLL and the switchover between PLL and OSC or not.

### 5.6.2. Switchover from OSC to PLL

Write 1 into F1 to trigger the hardware to turn on PLL automatically and switch the clock sources of Clock 1 from OSC output frequency to PLL output frequency. After PLL is turned on by the hardware automatically, its output frequency is 6.5536MHz. Write 1 into F1 to complete the switch.

### 5.6.3. Switchover from PLL to OSC and Turning off Clocks

When PWRUP=0 (in the register AFSIG), write 1 into F0 to trigger the automatic switchover of Clock 1 from PLL to OSC by hardware, turn off PLL, and turn off Clock 1.

# 6. Power Consumption

## 6.1. System Status

According to the state of Clock 1 and Clock 2, the states of the system can be categorized into three modes:

- OSC Mode after Level 1 and Level 2 resets;
- Work Mode: Turn on PLL, and switch the system clock source to PLL clock;
- Sleep Mode: Switch the system clock source to OSC clock when PWRUP=0, and turn off the clock. It can be categorized into Sleep Mode and Deep Sleep Mode. In the Sleep Mode, the chip can recover to the OSC Mode when the Recovery Reset, IO Wake-up Reset, RST Pin Reset, or RTC Wake-up Reset occurs. In the Deep Sleep Mode, the chip can recover to the OSC Mode when POR/BOR, Recovery Reset or RST Pin Reset occurs.

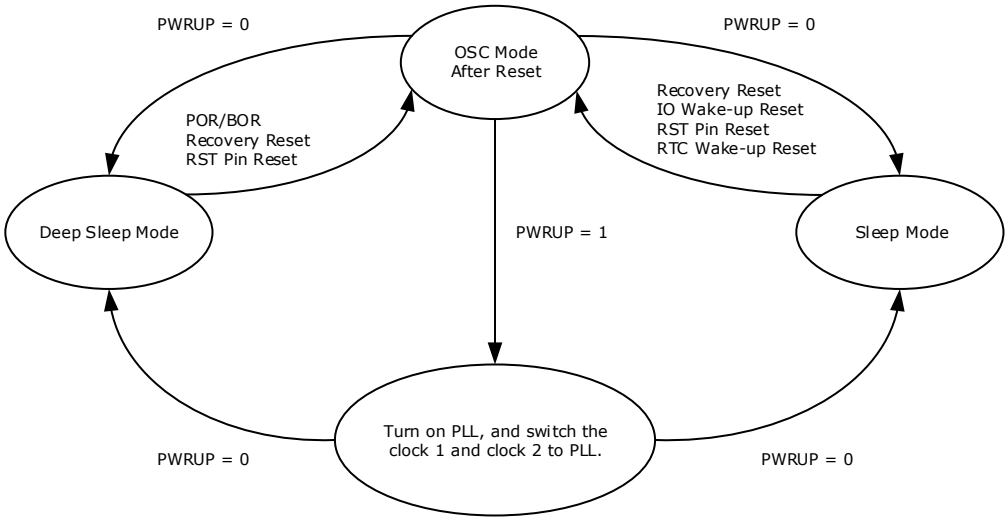



**Figure 6-1 System Status Switchover**

After reset, the reset reasons can be detected via querying the register Wake-up/ Reset Reason Query Register.

Figure 6-2 shows the relationship of the input voltage on the pin VDP, and the flags PWRUP and PWRDN.

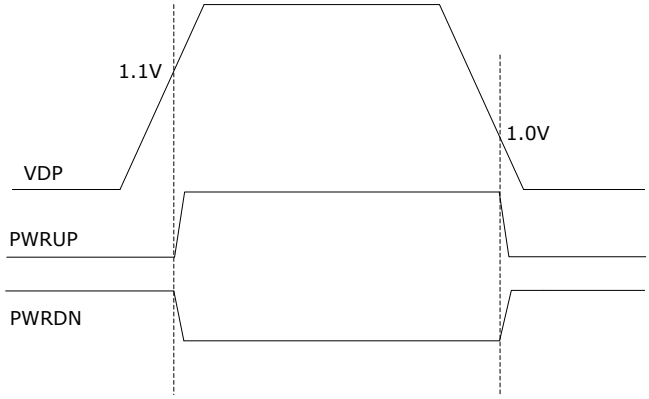



**Figure 6-2 Relationship of Input Voltage on Pin VDP, and Flags PWRUP and PWRDN**

In V9003/V9103, some modules can be turned off, but some cannot. And, some modules consumes power depending on the working voltage, and some depending on the working frequency. Table 6-1 lists all factors that can affect the power consumption of the modules.

**Table 6-1 Factors Affecting the Power Consumption of Every Module**

| Module | The status when powered on | Can be turned off? | Related to the clock frequency? | Related to LDO25? |
|---|---|---|---|---|
| LDO33 | On | When VDP<1.0V, and 1 is written into the bit LDO33PD (bit5, LDOCFG, 0x1B14), the LDO33 regulator is turned off. | No | No |
| LDO25 | On | No | No | No |
| OSC | On | No. Lower to 0.5μA. | No | No |
| MCU | On | Standby | Yes | Yes |
| Reference (Low Power Consumption) | On | No | No | No |
| RTC | On | No | No | No |
| PLL | Off | Yes | No | No |
| BGP | Off | Yes | No | No |

| | | | | |
|---|---|---|---|---|
| Power-down Monitoring Circuit | On | No | No | No |
| Temperature Measurement Circuit | Off | Yes | No | No |
| Battery Voltage Measurement Circuit | Off | Yes | No | No |
| LCD | The driver is off, and the scanning timing is on. | Yes | No. Relative to the driver. | No |
| ADC | No | Yes | Yes | Yes |
| Energy Metering Circuit | The channels are off, and the metering timing is on. | No | Yes | No |

To lower the power consumption of the system, there are three operation modes:

- Normal mode
- Low-power-consumption mode
- Sleep mode

# 6.2. Default Status

The default status of the system is as follows.

LDO33 is on, OSC is on, LDO25 is on, the output voltage of LDO25 is 2. 5V, and, CPU is on.

**Table 6-2 Default Status of the System**

| Module | Status when powered on | Can be turned off? | Current status | Power Consumption (μA) |
|---|---|---|---|---|
| LDO33 | On | When VDP<1.0V, and 1 is written into the bit LDO33PD (bit5, LDOCFG, 0x1B14), the LDO33 regulator is turned off. | On | 350 |
| LDO25 | On | No | On | 1.5 |

| | | | | |
|---|---|---|---|---|
| Reference (Low Power Consumption) | On | No | On | 1 |
| OSC | On | No. The power consumption can be lowered to 0.5μA. | On | 2~3 |
| RTC | On | No | On | |
| MCU | On | Standby | On | 1100 |
| Total | About 1455μA | | | |

# 6.3. Normal Mode

In the OSC Mode, turn on PLL, switch the clock sources of Clock1 and Clock2 from OSC clock to PLL clock, and the chip gets into working mode. In the Working Mode, select the appropriate PLL clock, turn on several ADC channels, corresponding metering channels and LCD drivers, and use other peripherals of CPU.

The power consumption of the chip depends on the PLL clock, the number of ADC channels and the configured output voltage of LDO25.

Table 6-3 shows the status and the power consumption of the modules when PLL is turned on, the clock frequency for MCU is 6.5536MHz, and the output voltage of LDO25 is 2.5V.

**Table 6-3 Power Consumption (PLL Frequency=6.5536MHz, Output voltage=2.5V)**

| Module | Status when powered on | Can be turned off? | Current status | Power Consumption (μA) |
|---|---|---|---|---|
| LDO33 | On | When VDP<1.0V, and 1 is written into the bit LDO33PD (bit5, LDOCFG, 0x1B14), the LDO33 regulator is turned off. | On | 350 |
| LDO25 | On | No | On | 1.5 |
| Reference (Low Power Consumption) | On | No | On | 1 |
| OSC | On | No. The power consumption can be lowered to 0.5μA via configuration. | On | 1 |

| | | | | |
|---|---|---|---|---|
| RTC | On | No | On | |
| PLL | Off | Yes | On | 30 |
| BGP | Off | Yes | On | 300 |
| Temperature Measurement | Off | Yes | Off | 60 |
| Battery Voltage Measurement | Off | Yes | Off | 130 |
| ADC Module | Off | Yes | On | 3000 |
| Energy Metering DSP | Channel, off; Clock, On | Yes | On | 4300 |
| LCD | Driver, off; Scanning Timing, on | Yes | On | 29 |
| MCU | On | Standby | On | 4300 |

**Table 6-4 Power Consumption of ADCs**

| Module | Operating Frequency | When | Default | Power Consumption (μA) |
|---|---|---|---|---|
| ADC | 6.5536MHz/8 | Current Channel IA | Off | 400 |
| | | Current Channel IB | | 400 |
| | | Current Channel IC | | 400 |
| | | Current Channel IN | | 400 |
| | | Voltage Channel UA | | 400 |
| | | Voltage Channel UB | | 400 |
| | | Voltage Channel UC | | 400 |
| | | Measurement Channel for voltage measurement | | 220 |

**Table 6-5 Power Consumption of the Metering Block**

| Module | Operating Frequency | Default | Power Consumption (μA) |
|---|---|---|---|
| Metering Block | 6.5536MHz | Digital metering circuits are turned on automatically after reset. The clock is turned on automatically. Other circuits are turned off. | 4300 |
| | 1.6384MHz | | 1100 |

When testing the LCD module, without a screen, the power consumption of the resistor ladder is independent of the scanning frequency; and, with a screen, higher scanning frequency consumes more power.

**Table 6-6 Power Consumption of LCD without A Screen**

| Module | Driver | When | Default | Power Consumption (μA) |
|---|---|---|---|---|
| LCD (Without a screen, all pixels are lit) | Resistor Ladder | 600 kΩ | Turn off the driver, and turn on the scanning timing. | 5 |
| | | 300 kΩ | | 11 |
| | | 180 kΩ | | 18 |
| | | 120 kΩ | | 27 |
| | Charge Pump | 64 Hz | | 0.5 |
| | | 128 Hz | | 0.5 |
| | | 256 Hz | | 0.5 |
| | | 512 Hz | | 0.5 |

The power consumption of LCD with a screen depends on the selected screen.

Table 6-7 shows the power consumption of the MCU.

**Table 6-7 Power Consumption of MCU**

| Block | Status | When | When | Default | Power Consumption (μA) |
|---|---|---|---|---|---|
| MCU | On | LDO25=2.23V | PLL=6553.6kHz | Off | 4300 |
| | | | OSC | On | 1100 |

# 6.4. Typical Power Consumption

**Table 6-8 Typical Power Consumption**

| LDO25=2.5V | Power Consumption (mA) |
|---|---|
| After reset, $f_{MCU}$=32.768kHz, turn off ADCs, PLL clock, LCD, and so on. | 1.1 |
| $f_{MCU}$=6.5536MHz, turn on all modules, and start metering energy | 13 |
| $f_{MCU}$=32.768kHz, , turn on all modules, and start metering energy | 9.8 |

# 6.5. Sleep Mode

## 6.5.1. Sleep/Wake-up Process

The system is controlled by software completely. Software queries the value of the flag PWRUP in the register AFSIG. If PWRUP=0 and the clock source for Clock 1 and Clock 2 is switched to OSC clock (CLKS=0), the system gets into sleep when SYSG=1, and the system gets into Deep Sleep or Sleep according to the value of the bit DEEP of the register DEEP.

In Sleep or Deep Sleep, RTC continues to run; all memories get into low-power consumption mode; except LCD and energy metering DSP, CPU and all peripherals stop working and get into low-power-consumption mode; LDO25 outputs 2.5V voltage; and write 1 into the bit LDO33PD (bit5, LDOCFG, 0x1B14) to turn off LDO33, and enable the input voltage to LDO33 equal to that on the pin VDD5, to lower power consumption. The lowest power consumption state is achieved if all ADCs, PLL clock, LCD driving/timing circuits, and energy metering DSP have been disabled or turned off, and IOs are configured to output and input disabled, before entering Sleep or Deep Sleep. If the LCD driver/timing circuit is on, display still can be performed normally in Sleep or Deep Sleep.

In Sleep, the Recovery Reset, RTC Wake-up Reset, RST Pin Reset and IO Wake-up Reset all can wake up the system. In Deep Sleep, only POR/BOR, Recovery Reset, or RST Pin Reset can wake up the system. To wake up the system via IO Wake-up Reset, it is necessary to set the pin WAKE0, WAKE1 or WAKE2 as input before getting into Sleep or Deep Sleep. A high-to-low transition (both the high level and low level need to last at least 4 OSC clock periods) on any one of these IO ports can cause IO Wake-up Reset, see "IO Wake-up Reset" for details.

The system is woken up and get back to the OSC Mode. Turn on PLL, and switch the clock sources of Clock 1 and Clock 2 to PLL clock, to enable the system working normally.

The start address of the system status control registers is 0x1200, and the bit width of them is 1-bit. To protect the system from the error operation, the write operation of the above registers should be done as follows.

1. Write the access password 0xD6 into the register located at 0x1217;

2. Write the access password 0xAD into the register located at 0x1218 to enable accessing to the clock control registers (0x1210~0x1215). Both write operations should be continuous;

3. Write the password 0101101 into Bit7~Bit1 of the system clock control registers (0x1210~0x1215), and then, write the appropriate value into the Bit0 of them.

4. Configure the system status via configuring the system status control registers;

5. After configuration, write 0xAE into the control register located at 0x1218 to protect the system clock control registers (0x1210~0x1215) from being access.

Read of the above registers is not protected. Bit7~Bit1 are read out as 0.

When all above configuration is done, if higher frequency is needed for the system, PLL must be turned on via configuring the register PLLCtrl. And, PLL must be turned off via configuring the register PLLCtrl, to ensure the low power consumption of the system.

## 6.5.2. Power Consumption in Sleep

Table 6-9 shows the power consumption in Sleep mode.

**Table 6-9 Power Consumption in Sleep**

| Module | Status when powered on | Can be turned off? | Power Consumption (μA) |
|---|---|---|---|
| LDO25 | On | No | 2.5 |
| Reference (Low Power Consumption) | On | No | |
| OSC | On | No. The power consumption can be lowered to 0.5μA. | 2.3 |
| RTC | On | No | |
| Electric leakage of the chip | | | 4 |
| Total | 8.8μA | | |

# 7. Power Management

## 7.1. Features

- Single power supply of 5V;
- The internal analog circuit and IOs powered by LDO33;
- The internal digital circuit and PLL powered by LDO25;
- Brown-out Reset, improving the reliability of the system;
- Low-voltage monitor, monitoring the battery voltage in real-time.

## 7.2. LDO33 Regulator

LDO33 supplies power to the internal analog circuit and IOs. If the external voltage is higher than 3.3V, the LDO33 outputs 3.3V voltage to protect the analog circuit performance from disturbance of 5V main supply (Figure 16-1). When the voltage on the pin VDP is lower than 1.0V, and the flag PWRUP in register AFSIG is set to 0, 1 is written into the bit LDO33PD (bit5, LDOCFG, 0x1B14) to turn off the LDO33 regulator, and enable the input voltage to LDO33 equal to that on the pin VDD5 to lower power consumption.

LDO33 can provide up to 30mA driving current. When the load current through the analog circuit and IOs is less than 30mA, the LDO33 output voltage is stable. If it is higher than 30mA, the increase of loading current leads to the decrease of the LDO33 output voltage (Figure 16-2).

Decoupling capacitors are needed to connect to the LDO33 outside the chip. 4.7μF and 0.1μF capacitors in parallel are recommended.

Configure the bit V3P5 of LDO Control Register (LDOCFG, Table 7-1) to supply 3.5V voltage to some special LCD screens (Table 7-2).

**Table 7-1 LDO Control Register (LDOCFG, 0x1B14)**

| 0x1B14, R/W | LDO Control Register, LDOCFG | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | Bit4 | bit3 | bit2 | bit1 | bit0 |
| | V3P | | LDO33PD | V3P5 | | | LDOVSEL<1> | LDOVSEL<0> |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 7-2 Bit Description of LDOCFG**

| Bit | Description | Remark |
|---|---|---|
| V3P | To select the LCD screen type | 0, 3.3V LCD; 1, 3V LCD |
| LDO33PD | To turn off LDO33 regulator | 1, to turn off LDO33 regulator (only valid when power-down occurs). |
| V3P5 | To select the output voltage of LDO33 | 0, 3.3V; 1, 3.5V |
| LDOVSEL<1:0> | To select the output voltage of LDO25 regulator | 00, 2.5V; 10, 2.75V;<br>01, 2.23V; 11, 2.57V |

# 7.3. LDO25 Regulator

LDO25 supplies power to the digital circuit and PLL. When the 5V main supply is 200mV higher than LDO25, LDO25 generates a 2.5V voltage, which is independent of the 5V main supply. LDO25 can provide up to 35mA driving current. When the loading current through the digital circuit is lower than 35mA, LDO25 output voltage is constant. If it is higher than 35mA, the output voltage of LDO25 decreases (Figure 16-3).

Unlike LDO33, LDO25 still can work in power-outage state, which can ensure the power supply to the digital circuit when the main supply sags.

Decoupling capacitors are needed to connect to the LDO25 outside the chip. 4.7μF and 0.1μF capacitors in parallel are recommended.

The output voltage of LDO25 is programmable. It can be configured to be 2.23V, 2.57V, 2.5V and 2.75V. Decrease the output voltage of LDO25 to lower the power consumption of digital circuit. Set the bits LDOVSEL<1> and LDOVSEL<0> in the register LDOCFG to configure the LDO25 output voltage. For example, when LDOVSEL<1> is configured to be 0, and LDOVSEL<0> is configured to be 1, LDO25 outputs 2.23V voltage, the lowest voltage level. The lower the LDO25 output voltage is, the lower power the digital circuit consumes.

After POR/BOR, RST Pin Reset or IO/RTC Wake-up Reset, LDO25 output voltage is 2.5V.

# 7.4. Power-down Monitoring Circuit

An embedded power-down monitoring module is used to monitor the input signal on the pin VDP in real-time. The battery voltage and 5V main supply are input into VDP after resistive division. When the voltage on VDP is lower than the reference voltage (about 1V), a power-down is detected, the MCU generates an interrupt, and the flag PWRDN in the register AFSIG is set. The power-down monitor module is always on.

# 7.5. POR/BOR Circuits

When the chip is powered on, the chip supplies two reset sources to ensure the proper reset. One of them is RST pin reset signal, and the other is POR signal. POR/BOR circuit monitors the level of 2.5V-LDO. The reset signal cannot be released and the system stays in reset status until the LDO25 output voltage is higher than 1.8V (a threshold).

In power-down, when the LDO25 level is lower than 1.7V, the BOR circuit generates a reset signal to reset the system.

This circuit can work all the time, even though the MCU is off.

# 7.6. BANDGAP Circuit

The BandGap circuit supplies reference voltage and current to ADC, PLL and power-down monitoring circuit. So, it should be turned on before them.

Load current and temperature have little effect on the reference voltage (Figure 16-4 and Figure 16-5 in Appendix).

This is an internal 1.185V Reference with a typical temperature drift of 20ppm/°C. There is some programmability associated with the 1.185V reference:

1. Configure BGPPDN in PLLCtrl to be 1 to turn on BandGap;
2. Configure the bit BGPCHOPN in the register BGPCFG (Table 7-4) to be 0 to enable BGPChop; When BGPChop is enabled, the direct current biasing can be removed, the output of the BandGap fluctuates over the range -50~+50mV, and the temperature coefficient can be improved by about 5ppm;
3. Adjust the REST<2:0> in the register BGPCFG (Table 7-4).

**Table 7-3 BGP Configuration Register (BGPCFG, 0x1B0E)**

| 0x1B0E,R/W | BGP Configuration Register, BGPCFG | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | REST<2> | REST<1> | REST<0> | | | | BGPCHOPN | |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 7-4 Global Current Bias Control Register (CURRCFG, 0x1B0F)**

| 0x1B0F,R/W | Global Current Bias Control Register, CURRCFG | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | | CURRLP | RTRIM<2> | RTRIM<1> | RTRIM<0> |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 7-5 Bit Description of BGPCFG and CURRCFG**

| Bit | Description | Remark |
|---|---|---|
| BGPCHOPN | To enable chop stabilization of BandGap | 0, On (default); 1, Off. |
| REST<2:0> | To adjust temperature coefficient of BandGap. | |
| CURRLP | Normal mode, or to adjust the global bias current to 1/3. | 0, normal mode;<br>1, low-power-consumption mode. |
| RTRIM<2:0> | To adjust global bias current over the range of ±15%. | 000, +15%; 001, +10%; 010, +5%; 011, 0%; 100, -5%; 01, -10%; 110, -15%; 111, unavailable |

# 7.7. Battery Discharge

V9003/V9103 can be powered on by batteries. When the input level on the pin VDP is lower than 1.0V, PWRUP=0 (bit2 of AFSIG), the 5.0V main supply is powered down, even the power supply is switched to battery supply which depends on the voltage drop on the two diodes in Figure 7-1.

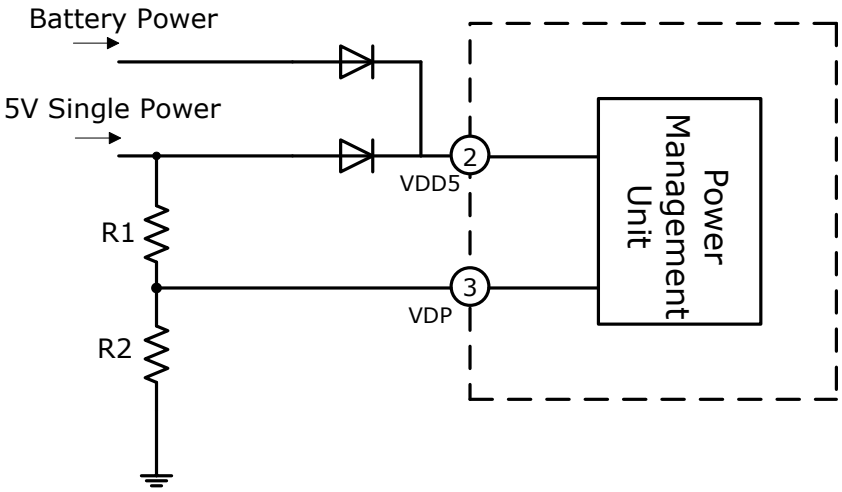



**Figure 7-1 Switchover between 5V Single Power and Battery**

The battery will get passive when it is not used for a long time. So users must set the bits BATDISC<1> and BATDISC<0> (Bit4 and Bit3 of CtrlBAT, 0x1B10) at intervals to discharge the batteries to protect them from passivation. When being discharged, the batteries consume 3mA.

Discharge the batteries in a short time to save power. After discharge, configure the bits BATDISC<1> and BATDISC<0> to 0.

**Table 7-6 Battery Discharge Control Register (CtrlBAT, 0x1B10)**

| 0x1B10, R/W | Battery Discharge Control Register, CtrlBAT | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | Bit4 | bit3 | bit2 | bit1 | bit0 |
| | | | | BATDISC<1> | BATDISC<0> | | | |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 7-7 Bit Description of BATDISC<1> and BATDISC<0>**

| Bit | | Description | Remark |
|---|---|---|---|
| Bit4 | BATDISC<1> | To enable battery discharge via the pin BAT1 | 1, to enable. By default, 0. |
| Bit3 | BATDISC<0> | To enable battery discharge via the pin BAT2 | 1, to enable. By default, 0. |

# 8. Energy Metering

## 8.1. Features

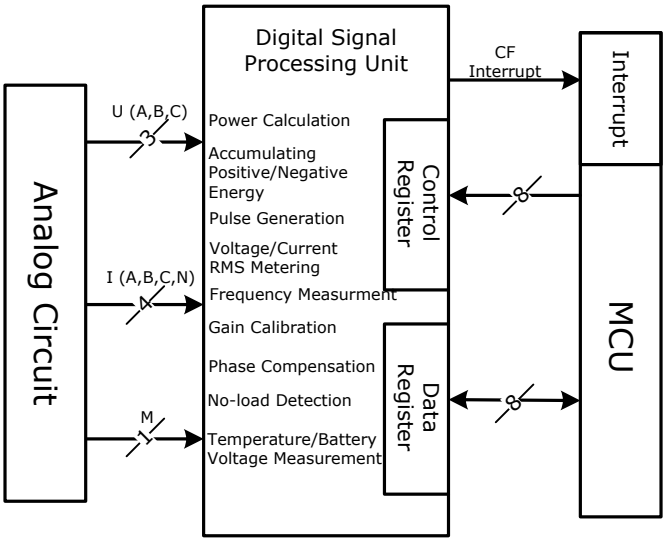



**Figure 8-1 Schematics of Energy Metering Block**

Figure 8-1 shows the schematics of energy metering module. The features of the energy metering circuit of V9003/V9103 are as follows:

- Eight high-performance oversampling Σ/ΔADC channels are available, including Voltage Channel UA/UB/UC, Current Channel IA/IB/IC/IN, and the M Channel for temperature, battery voltage and external signals measurement;
- Can be configured into 3-phase meters for both 3-wire and 4-wire service;
- The active energy, reactive energy, fundamental active energy, and fundamental reactive energy of every phase and on the overall system can be metered;
- The positive/negative active energy of every phase and on the overall system can be metered;
- The active energy can be metered with error less than 0.1% within the dynamic range of 2000:1;
- The reactive energy can be metered with error less than 0.1% within the dynamic range of 1000:1;

- Current RMS, voltage RMS, and fundamental current/voltage RMS can be metered;
- The DC component in every channel can be metered;
- Power factor angle of every phase are available;
- Frequency and phase measurement is available;
- No-load detection is available, and the no-load detection threshold is configurable;
- The current through the phase wires and neutral wire is monitored continuously to detect the fault of the power distribution system;
- Active power gain can be calibrated in 3 segments;
- Phase error is configurable over a range of ±2.8 degrees, which can be compensated in 5 segments;
- Two configurable CF pulse outputs and CF interrupts are available;
- Meter up to the 31$^{st}$ harmonic.

Figure 8-2, Figure 8-3, and Figure 8-4 show how the signals are processed.

1. As shown in Figure 8-2, the amplified analog signals from APGAs are converted into digital signals, and then they are filtered in the decimation filters to get DC and AC components of signals;
2. The AC components of M Channel and Current Channel IN are processed as shown in Figure 8-3. Only RMS calculation is necessary;
3. Take Phase A as an example to show how the AC components of the current channels and voltage channels of Phase A/B/C are processed (Figure 8-4):
   - To calculate the RMS values of UA and IA;
   - To calculate the RMS values of the fundamental component of UA and IA;
   - To calculate the active power of Phase A, and, to accumulate the active energy;
   - To calculate the fundamental active power of Phase A, and, to obtain the fundamental active energy;
   - To increment the relative phase angle of current and voltage in Phase A by 90 Degree via the Hilbert Filter;

- To calculate the reactive power of Phase A, and, to obtain the reactive energy;
- To calculate the fundamental reactive power of Phase A, and, to obtain the fundamental reactive energy.

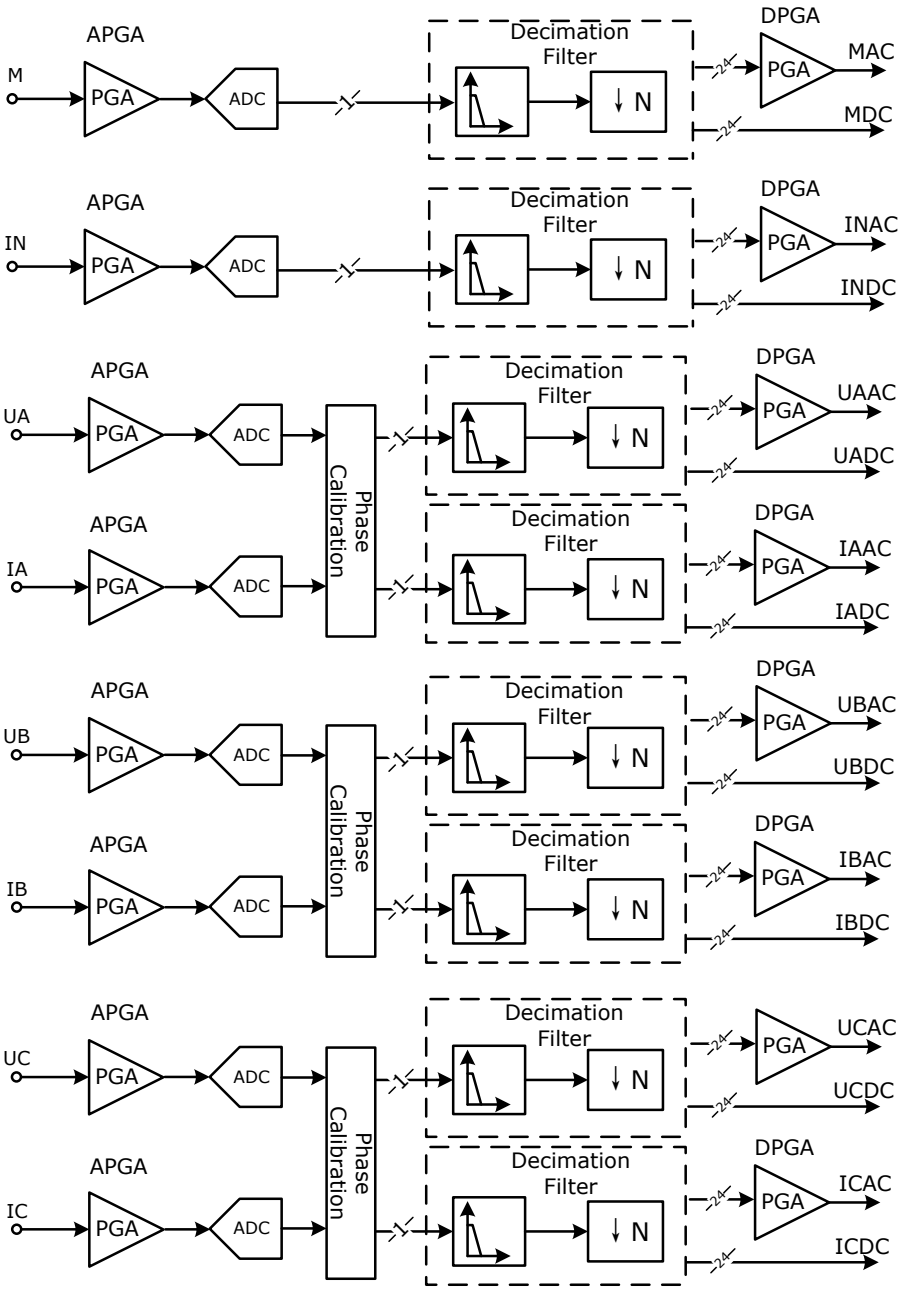



**Figure 8-2 Processing Signals 1**

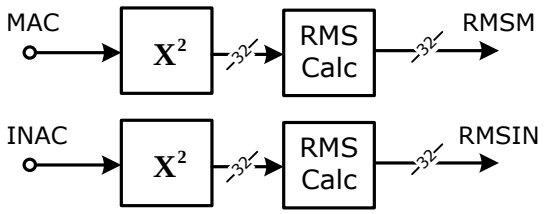



**Figure 8-3 Processing Signals 2**

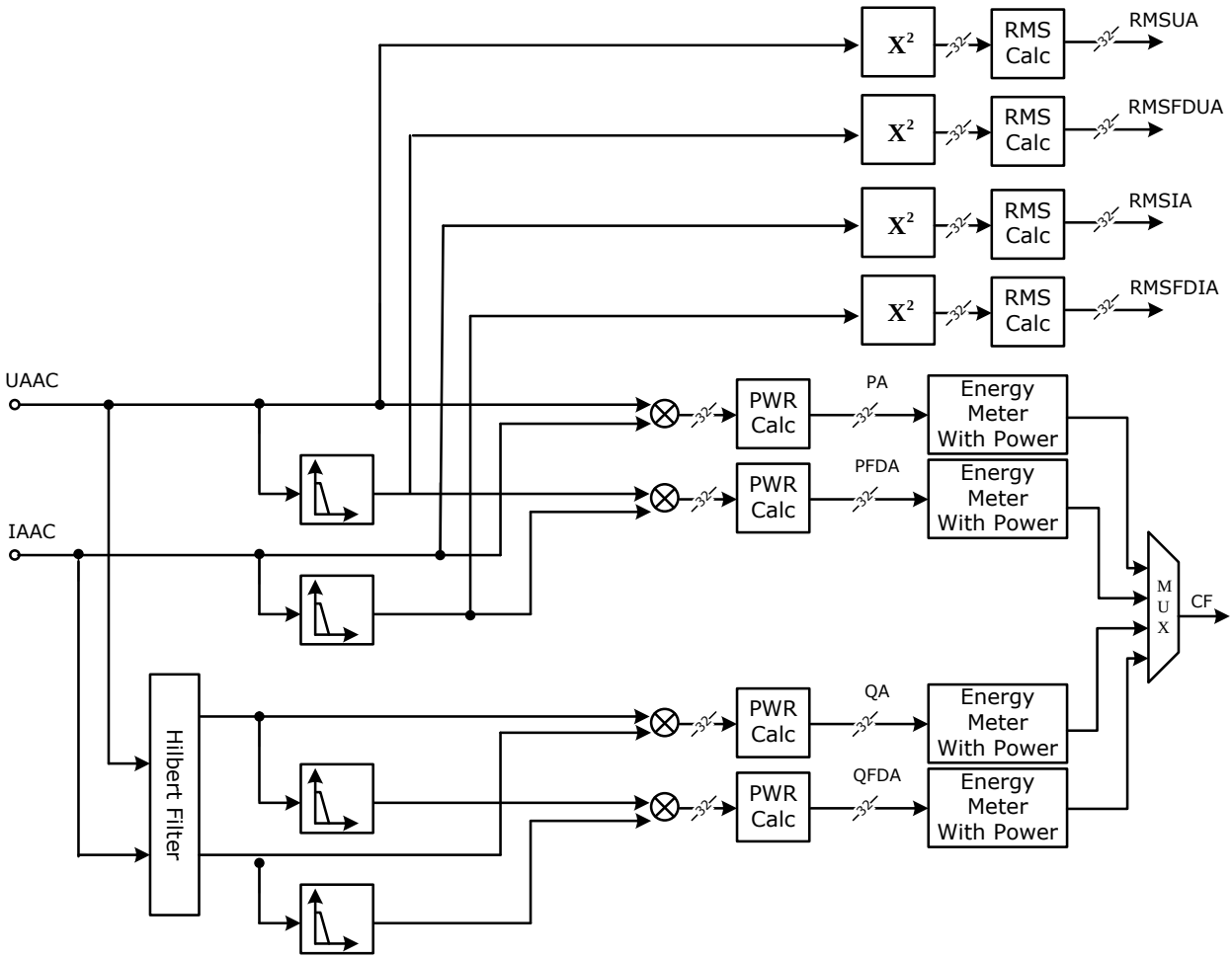



**Figure 8-4 Processing Signals 3**

# 8.2. Access to Energy Metering Registers

In V9003/V9103, all the energy metering registers are accessed by MCU asynchronously. MCU only can write of and read the 32-bit energy metering registers via the metering access registers.

Table 8-1 lists metering access registers.

**Table 8-1 Metering Access Registers**

| No. | Address | Mnemonic | Description |
|---|---|---|---|
| 1 | 0x3000 | RBUF3 | The highest byte [31:24] of the read buffer register |
| 2 | 0x3001 | RBUF2 | The second highest byte [23:16] of the read buffer register |
| 3 | 0x3002 | RBUF1 | The second lowest byte [15:8] of the read buffer register |
| 4 | 0x3003 | RBUF0 | The lowest byte [7:0] of the read buffer register |
| 5 | 0x3004 | WBUF3 | The highest byte [31:24] of the write buffer register |
| 6 | 0x3005 | WBUF2 | The second highest byte [23:16] of the write buffer register |
| 7 | 0x3006 | WBUF1 | The second lowest byte [15:8] of the write buffer register |
| 8 | 0x3007 | WBUF0 | The lowest byte [7:0] of the write buffer register |
| 9 | 0x3008 | FREADY | Communication is done. |

Reading data:

1. MCU accesses the addresses 0x2000~0x2FFF via MOVX instructions, and reads the energy metering data in the corresponding addresses into the access module;
2. MCU accesses the flag FREADY (0x3008) via MOVX instructions. When it is read out as 1, the access module gets the energy metering data.
3. MCU reads out the data from the 32-bit data buffer register (0x3000~0x3003) via MOVX instruction to finish read operation.

Writing data:

1. MCU writes of 32-bit data buffer register (0x3004~0x3007) via MOVX instruction;
2. MCU writes of 0x2000~0x2FFF via MOVX instruction, and writes the data in the buffers into the corresponding addresses via the access module;
3. MCU accesses the flag FREADY (0x3008) via MOVX instructions. When it is read out as 1, the data is written into the corresponding addresses via the access module.

Attentions: If DSP and MCU access the energy metering module synchronously, DSP has the priority to process the digital signals, and MCU accesses the flag FREADY (0x3008) via MOVX instruction continuously when the processing. When the flag is read out as 1, the data has been written into the corresponding address, or the data has been read out from the corresponding address.

To protect the metering parameters from mis-configuration, the writing and reading the metering access registers are controlled by the register PMPwd (0x1B30).

**Table 8-2 Metering Access Control Register (PMPwd, 0x1B30)**

| 0x1B30, R/W | Metering Access Control Register, only Bit0 can be read out, PMPwd | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Function | | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| Disable Access Protection | 0xCD | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 |
| Enable Access Protection | 0xCC | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| Default Value | 0x00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# 8.3. Energy Metering Registers

## 8.3.1. Metering Configuration Registers

There are 3 32-bit registers (MTPARA0~MTPARA2) in the metering register bank of V9003/V9103. They can provide 96-bit control bits to configure meter type, parameters, control status and on/off state, and so on.

**Table 8-3 Metering Configuration Registers**

| No. | Address | Register | Description |
|---|---|---|---|
| 1 | 0x2000 | MTPARA0 | Metering Configuration Register 0 |
| 2 | 0x2001 | MTPARA1 | Metering Configuration Register 1 |
| 3 | 0x2002 | MTPARA2 | Metering Configuration Register 2 |

Table 8-4 lists all control information in the metering configuration registers.

**Table 8-4 Instruction of Metering Configuration Registers**

| | No. | Register | Width | Bit | Description |
|---|---|---|---|---|---|

| | | | (bit) | | |
|---|---|---|---|---|---|
| MTPARA0 | 1 | ADCON | 8 | [7:0] | To enable/disable signal input into ADCs.<br>1, to enable; 0, to disable<br>Bit7, UM; Bit6, IC; Bit5, IB; Bit4, IA; Bit3, IN; Bit2, UC; Bit1, UB; Bit0, UA |
| | 2 | MSKA | 5 | [12:8] | Return Difference Amplitude Control Register of Phase Compensation in Segments |
| | 3 | MSKP | 5 | [20:16] | Return Difference Amplitude Control Register of Gain Calibration in Segments |
| | 4 | SLPMPA | 1 | [24] | MPA goes into Sleep; The state machine stops working; RAM control is released. |
| | 5 | CLRMPA | 1 | [25] | MPA is initialized; RAM is cleared automatically. |
| | 6 | SLPMPB | 1 | [26] | MPB goes into Sleep; The state machine stops working; RAM control is released. |
| | 7 | CLRMPB | 1 | [27] | MPB is initialized; RAM is cleared automatically. |
| | 8 | SLPMPC | 1 | [28] | MPC goes into Sleep; The state machine stops working; RAM control is released. |
| | 9 | CLRMPC | 1 | [29] | MPC is initialized; RAM is cleared automatically. |
| | 10 | SLPHLB | 1 | [30] | HLB goes into Sleep; The state machine stops working; RAM control is released. |
| | 11 | CLRHLB | 1 | [31] | HLB is initialized; RAM is cleared automatically. |
| MTPARA1 | 1 | ATXUA | 3 | [2:0] | Digital gains in Channel UA, 2^atx_ua, (1~128) |
| | 2 | ATXIA | 3 | [6:4] | Digital gains in Channel IA, 2^atx_ia, (1~128) |
| | 3 | ATXUB | 3 | [10:8] | Digital gains in Channel UB, 2^atx_ub, (1~128) |
| | 4 | ATXIB | 3 | [14:12] | Digital gains in Channel IB, 2^atx_ib, (1~128) |
| | 5 | ATXUC | 3 | [18:16] | Digital gains in Channel UC, 2^atx_uc, (1~128) |
| | 6 | ATXIC | 3 | [22:20] | Digital gains in Channel IC, 2^atx_ic, (1~128) |
| | 7 | ATXIN | 3 | [26:24] | Digital gains in Channel IN, 2^atx_in, (1~128) |
| | 8 | ATXM | 3 | [30:28] | Digital gains in M Channel, 2^atx_um, (1~128) |
| MTPARA2 | 1 | CFON | 2 | [1:0] | To turn on/off CF2 and CF1, respectively. |

| | | | | |
|---|---|---|---|---|
| | | | | Bit0, CF1; Bit1, CF2. 1, on; 0, off. |
| 2 | MTMODE | 2 | [3:2] | 0, for 3-phase, 4-wire distribution system<br>1, for 3-phase, 3-wire distribution system (only energy in Phase A and Phase C is metered)<br>2, N/A; 3, N/A |
| 3 | CFSEL | 4 | [7:4] | To select the source of CF1/CF2 pulse.<br>0, to generate CF pulses based on the vector sum of active and reactive energy on the overall system;<br>1, to generate CF pulses based on the absolute sum of active and reactive energy on the overall system;<br>2, to generate CF pulses based on the active and reactive energy of Phase A;<br>3, to generate CF pulses based on the active and reactive energy of Phase B;<br>4, to generate CF pulses based on the active and reactive energy of Phase C;<br>5, to generate CF pulses based on the vector sum of fundamental active and reactive energy on the overall system;<br>6, to generate CF pulses based on the absolute sum of fundamental active and reactive energy on the overall system;<br>7, to generate CF pulses based on the fundamental active and reactive energy of Phase A;<br>8, to generate CF pulses based on the fundamental active and reactive energy of Phase B;<br>9, to generate CF pulses based on the fundamental active and reactive energy of Phase C. |
| 4 | AECEN | 1 | [8] | To disable phase compensation in segments. |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | 1, to disable; 0, to enable |
| | 5 | PECEN | 1 | [9] | To disable gain calibration in segments.<br>1, to disable; 0, to enable |
| | 6 | CRPEN | 1 | [10] | To disable anti-creeping.<br>1, to disable; 0, to enable |
| | 7 | CLRACC | 1 | [11] | To clear the energy tanks.<br>1, to clear; 0, not to clear |

## 8.3.2. Metering Control and Data Registers

**Table 8-5 Metering Control and Data Registers**

| Registers | Start Address | Width (bit) | Description |
|---|---|---|---|
| DC component data registers | 0x2812 | 32 | See "Decimation Filter" for details |
| Voltage/current RMS registers | 0x2A4F | 32 | See "Calculating Voltage and Current RMS" for details |
| Voltage/current RMS gain registers | 0x2AAC | 32 | |
| Active/reactive power data registers | 0x2C01 | 32 | See "Calculating Power" for details |
| Active power gain registers | 0x2A9D | 32 | |
| Active power gain calibration in segments related registers | 0x2ACC | 32 | |
| Apparent power registers | 0x2C95 | 32 | |
| Calibration in segments and no-load detection checkup register | 0x200E | 32 | |
| Power factor angle registers | 0x2C9D | 32 | |
| Phase compensation data registers | 0x2A98 | 32 | See "Phase Compensation" for details |
| Threshold registers for phase compensation in segments | 0x2AC0 | 32 | |
| Energy threshold registers and no-load detection threshold register | 0x2CA6 | 32 | See "No-load Detection" for details |

| | | | |
|---|---|---|---|
| Energy tanks and CF pulse counters | 0x2C09 | 32 | See "Metering Energy" for details |
| Frequency-phase data registers | 0x2008 | 32 | See "Frequency-phase Measurement" for details |

# 8.4. Channel Control

After POR, all channels are off. Turn on PLL clock and switch the clock source to PLL clock, and then, turn on the analog circuit, enable the signal input, and so on.

Turn on the analog circuit, and input the signals into the DSP as follows:

1. Configure the bit BGPPDN in the register PLLCtrl (0x1B12) to turn on BGP;
2. Configure the register ADCCtrl (Table 8-6 to turn on ADCs to start analog-digital conversion. Every ADC can be turned on independently. So, the channel can be selected according to the chip type and application, to lower the power consumption.

**Table 8-6 ADC Control Register (ADCCtrl, 0x1B11)**

| 0x1B11, R/W | ADC Control Register, ADCCtrl | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | ADPDMN | ADPDUCN | ADPDUBN | ADPDUAN | ADPDINN | ADPDICN | ADPDIBN | ADPDIAN |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Table 8-7 lists the definition of every bit.

**Table 8-7 Bit Description of ADCCtrl**

| Bit | Function | Description |
|---|---|---|
| ADPDMN | To turn on/off M Channel ADC. | 0, off; 1, on |
| ADPDUCN | To turn on/off Voltage Channel UC ADC. | 0, off; 1, on |
| ADPDUBN | To turn on/off Voltage Channel UB ADC. | 0, off; 1, on |
| ADPDUAN | To turn on/off Voltage Channel UA ADC. | 0, off; 1, on |
| ADPDINN | To turn on/off Current Channel IN ADC. | 0, off; 1, on |
| ADPDICN | To turn on/off Current Channel IC ADC. | 0, off; 1, on |
| ADPDIBN | To turn on/off Current Channel IB ADC. | 0, off; 1, on |
| ADPDIAN | To turn on/off Current Channel IA ADC. | 0, off; 1, on |

3. Enable signal inputting into ADCs. When ADCs are on, the signal input into ADCs via configuration is disabled, and the input signal of the corresponding ADC is set to 0.

**Table 8-8 ADC ON/OFF Control Register, ADCON**

| Register | Width | Location | Description |
|---|---|---|---|
| ADCON | 8 | Bit[7:0] of MTPARA0 | To enable/disable signal input into ADCs.<br>1, to enable; 0, to disable<br>Bit7, UM; Bit6, IC; Bit5, IB; Bit4, IA; Bit3, IN; Bit2, UC; Bit1, UB; Bit0, UA |

The register ADCON is a part of the 32-bit Metering Configuration Register 0 (MTPARA0). The global MTPARA0 will be accessed together.

# 8.5. Gain Configuration

There are eight ADC channels, including 3 voltage channels, 3 current channels, a current channel of the neutral wire, and the measurement channel (M Channel). Every channel has analog PGA and digital PGA, and the total gain is determined by both of them.

Gain of 1 is the default value. When adjusting the gains, the analog gain is recommended, and the maximum output signal after PGAs (analog and digital) should not be over 1.2V.

**Table 8-9 Analog PGA Related Registers**

| No. | Address | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | R/W | Register | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| 2 | 0x1B00 | AGIA | | | | | | GIA<2> | GIA<1> | GIA<0> |
| 3 | 0x1B01 | AGIB | | | | | | GIB<2> | GIB<1> | GIB<0> |
| 4 | 0x1B02 | AGIC | | | | | | GIC<2> | GIC<1> | GIC<0> |
| 5 | 0x1B03 | AGIN | | | | | | GIN<2> | GIN<1> | GIN<0> |
| 6 | 0x1B04 | AGUA | | | | | | | | GUA |
| 7 | 0x1B05 | AGUB | | | | | | | | GUB |
| 8 | 0x1B06 | AGUC | | | | | | | | GUC |
| 9 | 0x1B07 | AGM | | | REFGM <1> | REFGM <0> | | | | GM |
| | Default | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Table 8-10 shows the definition of every bit.

**Table 8-10 Bit Description of Analog PGA Related Registers**

| Bit | Function | Description |
|---|---|---|
| GIA<2:0> | Gain control of Channel IA | 000, × 1; 001, × 2; 010: × 4; 011: × 8; 100/101/110/111: ×16 |
| GIB<2:0> | Gain control of Channel IB | 000, × 1; 001, × 2; 010: × 4; 011: × 8; 100/101/110/111: ×16 |
| GIC<2:0> | Gain control of Channel IC | 000, × 1; 001, × 2; 010: × 4; 011: × 8; 100/101/110/111: ×16 |
| GIN <2:0> | Gain control of Channel IN | 000, × 1; 001, × 2; 010: × 4; 011: × 8; 100/101/110/111: ×16 |
| GUA | Gain control of Channel UA | 0, ×1 (by default) |
| GUB | Gain control of Channel UB | 0, ×1 (by default) |
| GUC | Gain control of Channel UC | 0, ×1 (by default) |
| REFGM <1:0> | Reference gain control of Channel M | 00, ×1; 01, ×2; 10/11, ×4 |

| | | |
|---|---|---|
| GM | Signal gain control of Channel M | 0, ×1; 1, ×2 |

* The reference gain in M Channel ADC can magnify the signals, but cannot improve the SNR of signal;

** The signal gain in M Channel ADC can magnify the signal and can increase the SNR of signal.

The digital PGA configuration is stored in the register MTPARA1 (0x2001), as shown in Table 8-11.

**Table 8-11 Digital PGA Related Registers**

| | No. | Register | Width (bit) | Bit | Description |
|---|---|---|---|---|---|
| MTPARA1 | 1 | ATXUA | 3 | [2:0] | Digital Gains in Channel UA, 2^atx_ua (1~128) |
| | 2 | ATXIA | 3 | [6:4] | Digital Gains in Channel IA, 2^atx_ia (1~128) |
| | 3 | ATXUB | 3 | [10:8] | Digital Gains in Channel UB, 2^atx_ub (1~128) |
| | 4 | ATXIB | 3 | [14:12] | Digital Gains in Channel IB, 2^atx_ib (1~128) |
| | 5 | ATXUC | 3 | [18:16] | Digital Gains in Channel UC, 2^atx_uc, (1~128) |
| | 6 | ATXIC | 3 | [22:20] | Digital Gains in Channel IC, 2^atx_ic (1~128) |
| | 7 | ATXIN | 3 | [26:24] | Digital Gains in Channel IN, 2^atx_in (1~128) |
| | 8 | ATXM | 3 | [30:28] | Digital Gains in M Channel, 2^atx_um (1~128) |

# 8.6. Phase Compensation

Theoretically, AC current and voltage should have the same frequency and phase angle (ω), namely,

$$i = I \times \sin(\omega t);$$

$$u = U \times \sin(\omega t).$$

But in practice, there is a phase difference between current and voltage in the chip, namely,

$i = I \times \sin(\omega t)$

$u = U \times \sin(\omega t + \delta)$

Where,

δ is the phase difference between current and voltage, which may diminish the energy metering.

There are two methods to eliminate the phase difference:

1. Adjusting the resistor-divider network and the RC filter circuit for voltage and current input;
2. Compensating the phase angle of one signal. Adjusting the delay chains to change the signal phase is recommended.

V9003/V9103 uses the latter method to eliminate the phase difference. Figure 8-5 shows the schematics of phase compensation. In the voltage channels and current channels, there are delay chains with a certain length. According to the phase leading or lagging, voltage or current is selected to be lagged, so the minimum resolution of phase compensation is the phase difference between two adjacent units in the delay chain, and the total compensated phase is the product of the minimum resolution and the length of the delay chain.

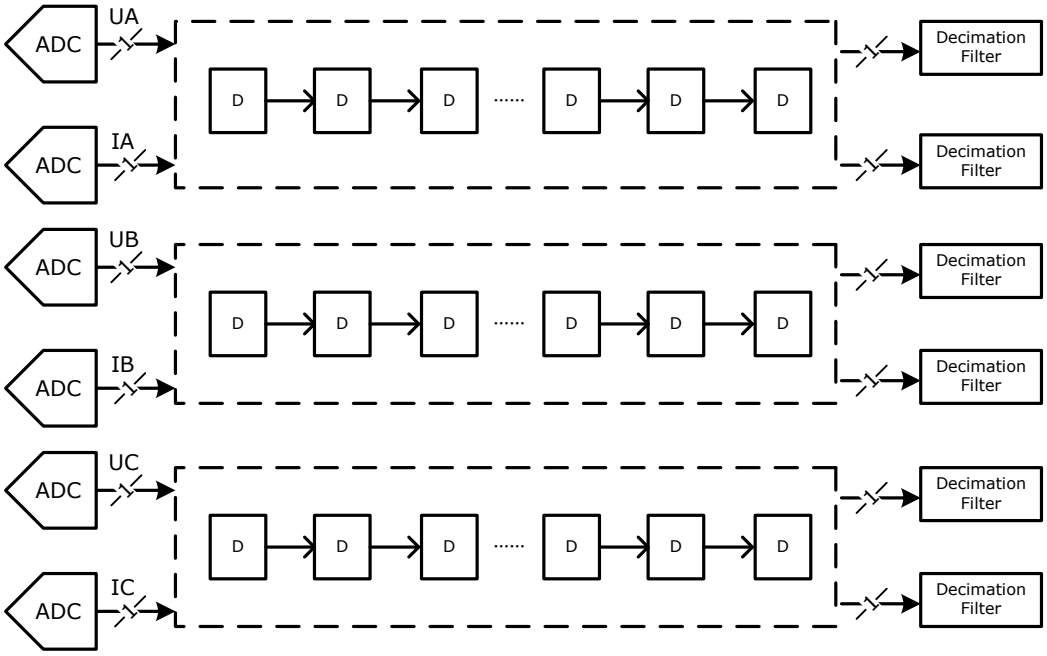



**Figure 8-5 Schematics of Phase Compensation**

Phase compensation is output as 1-bit digital bit stream.

The frequency of input signal in ADC channels is 50×16384, meaning that there are 16384 sampling points in a period. The phase of every sampling point should be compensated, so the smallest step length for compensation is 0.022 degrees. Because the register chain length is 128, the biggest delayed phase is 2.8 degrees, and the phase compensation can be configurable over the range of ±2.8 degrees.

To widen the dynamic range, the phase errors are designed to be compensated in segments. As shown in Figure 8-6, the chip is designed to compensate phase in 5 segments. Users can compare the calibrated current RMS to the preset current threshold in segments, and select the corresponding data register for phase compensation automatically.

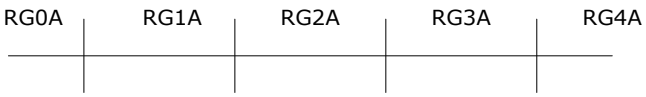



**Figure 8-6 Phase Compensation in Segments**

The return difference is designed, the amplitude of which is decided by Return Difference Amplitude Control Register of Phase Compensation in Segments (MSKA, Table 8-12). This register is 5-bit width, located at bit8~bit12 of the register MTPARA0, sized from 0 to 31, which decides the return difference.

**Table 8-12 Return Difference Amplitude Control Register of Phase Compensation in Segments, MSKA**

| Register | Address | Width | Location | Description |
|---|---|---|---|---|
| MSKA | 0x2000 | 5 | Bit[12:8] of MTPARA0 | Return Difference Amplitude Control Register of Phase Compensation in Segments |

For example, if the register MSKA is set to 4, it means that the last 4 bits of the current threshold register of the phase compensation in segments determine the return difference. So, if the threshold register , for example, AEC0A), is set to 0xabcd, the higher return difference point of the threshold, THH, is 0xabd0, and the lower return difference point of the threshold, THL, is 0xabc0.

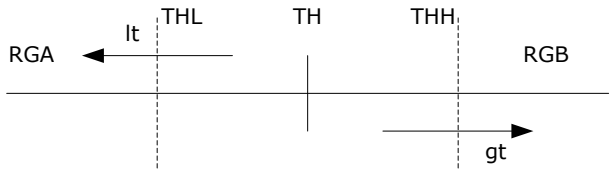



**Figure 8-7 Return Difference of the Phase Compensation**

Table 8-13 lists all phase compensation data registers in segments and the current threshold registers for phase compensation in segments. The phase compensation data registers in segments is 32-bit width, and the lower 3 bytes stores the total 24-bit phase compensation values of three channels. All registers can be read and written.

**Table 8-13 Phase Compensation Data Registers and Threshold Registers for Phase Compensation in Segments**

| No. | Address | Register | Description | |
|---|---|---|---|---|
| 1 | 0x2A98 | AEC0 | Segment 0 | Phase compensation in segments<br>Phase A: AECA=AECx[7:0]<br>Phase B: AECB=AECx[15:8]<br>Phase C: AECC=AECx[23:16]<br>Where, bit7/bit15/bit23 is the sign bit of the compensation value: 1, negative; 0, positive. And bit0~bit6/bit8~bit14/bit16~bit22 store the delay time of each phase, the absolute value of the compensation value, over the range of 0~127 (0~2.8 degrees).<br>By default, bit31~bit24 are set to 0. |
| 2 | 0x2A99 | AEC1 | Segment 1 | |
| 3 | 0x2A9A | AEC2 | Segment 2 | |
| 4 | 0x2A9B | AEC3 | Segment 3 | |
| 5 | 0x2A9C | AEC4 | Segment 4 | |
| 6 | 0x2AC0 | AEC0A* | Threshold 0 of phase compensation in segments of Current Channel IA | |
| 7 | 0x2AC1 | AEC0B | Threshold 0 of phase compensation in segments of Current Channel IB | |
| 8 | 0x2AC2 | AEC0C | Threshold 0 of phase compensation in segments of Current Channel IC | |
| 9 | 0x2AC3 | AEC1A* | Threshold 1 of phase compensation in segments of Current Channel IA | |
| 10 | 0x2AC4 | AEC1B | Threshold 1 of phase compensation in segments of Current Channel IB | |
| 11 | 0x2AC5 | AEC1C | Threshold 1 of phase compensation in segments of Current Channel IC | |
| 12 | 0x2AC6 | AEC2A* | Threshold 2 of phase compensation in segments of Current Channel IA | |
| 13 | 0x2AC7 | AEC2B | Threshold 2 of phase compensation in segments of Current Channel IB | |
| 14 | 0x2AC8 | AEC2C | Threshold 2 of phase compensation in segments of Current Channel IC | |
| 15 | 0x2AC9 | AEC3A* | Threshold 3 of phase compensation in segments of Current Channel IA | |
| 16 | 0x2ACA | AEC3B | Threshold 3 of phase compensation in segments of Current Channel IB | |
| 17 | 0x2ACB | AEC3C | Threshold 3 of phase compensation in segments of Current Channel IC | |

* The calibrated current RMS (IRMS) is used to determine which segment should be used. Take Phase A for example, when IRMS<AEC0A, the register AEC0 is used; when AEC0A<IRMS<AEC1A, the register AEC1 is used; when AEC1A<IRMS<AEC2A, the register AEC2 is used; when AEC2A<IRMS<AEC3A, the register AEC3 is used; when IRMS>AEC3A, the register AEC4 is used.

Use Phase Compensation in Segments Control Register (AECEN, Table 8-14) to enable or disable

compensating phase in segments. When phase compensation in segments is disabled, the register AEC0 is used.

**Table 8-14 Phase Compensation in Segments Control Register, AECEN**

| Register | Address | Width | Location | Description |
|---|---|---|---|---|
| AECEN | 0x2002 | 1 | Bit8 of MTPARA2 | Phase Compensation in Segments Control Register. 1, to disable; 0, to enable. |

# 8.7. Decimation Filter

What phase compensation does is to finely adjust the phase of the oversampling Σ-ΔADC, but not to change the signal spectrum. The oversampling Σ-ΔADC outputs 1-bit bit stream, followed by a large amount of noise with high frequency, which can be restrained by a decimation filter. The sampling frequency of the system clock can be lowered to FCLK/N from FCLK via filtering, where N is the decimation ratio of the filter, and the data length is extended to more bits from 1 bit.

As for V9003/V9103, N=128, so the sampling frequency after filtering is FCLK/128, and 256 22-bit sampling points per period is obtained, which is the instantaneous data.

The instantaneous data is composed of AC component and DC component. When it is filtered via the decimation filter (N=256), the DC component is obtained.

**Table 8-15 DC Component Data Registers**

| No. | Address | Register | Description | |
|---|---|---|---|---|
| 1 | 0x2812 | DCUA | DC component of Channel UA | The DC component is updated once per 20ms, and is stabilized in 80ms. |
| 2 | 0x2813 | DCIA | DC component of Channel IA | |
| 3 | 0x2814 | DCUB | DC component of Channel UB | |
| 4 | 0x2815 | DCIB | DC component of Channel IB | |
| 5 | 0x2816 | DCUC | DC component of Channel UC | |
| 6 | 0x2817 | DCIC | DC component of Channel IC | |
| 7 | 0x2818 | DCIN | DC component of Channel IN | |
| 8 | 0x2819 | DCM | DC voltage component of the M Channel | |

# 8.8. Hilbert Filter

The Hilbert Filter can increase the phase difference of the input signals by 90 degrees, which cannot be affected by the signal frequency. So, the Hilbert Filter is used to meter the reactive power.

The current and voltage signals are input into a Hilbert Filter, and then, the filtered signals multiply each other to get the instantaneous reactive power.

# 8.9. Obtaining Fundamentals

In the 50Hz power grid, some even harmonics, such as 3rd, 5th, and 7th harmonic, are transferred in the power system together with the 50Hz signal. In V9003/V9103, the input signal is low-pass filtered and downsampled via the decimation filter to get 4-point periodic waves for calculating fundamental power, fundamental energy, and fundamental RMS values.

The output signals from the Hilbert filter is also input to calculate the fundamental reactive power.

# 8.10. Calculating Power

Figure 8-8 shows the relationship between active power, reactive power, apparent power and power factor angle.

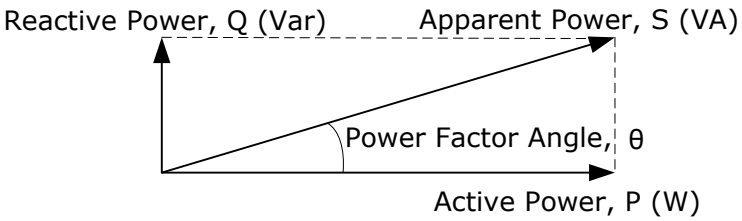



**Figure 8-8 Relationship between Active Power, Reactive Power, Apparent Power and Power Factor Angle**

## 8.10.1.Active and Reactive Power Calculation

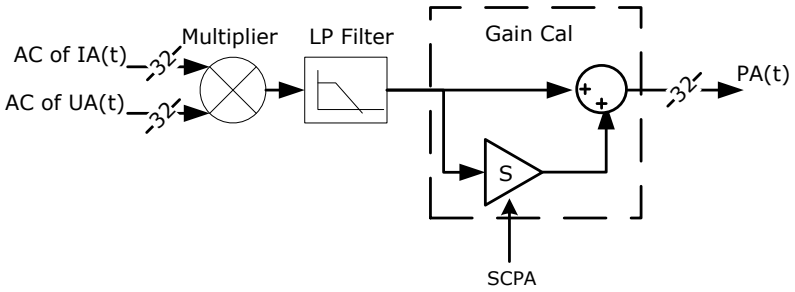



**Figure 8-9 Schematics of Power Calculation in Phase A**

Take Phase A for an example, the active and reactive power is calculated as shown in Figure 8-9. The instantaneous values of voltage and current are multiplied in the multiplier to get the instantaneous power. After LP Filter, the average power (P') is obtained, which is updated once per 20ms. After the gain calibration, use the following equation to get the power value.

$$Power = (1+S) \times P'$$

There are 12 power data in V9003/V9103, including active, reactive, fundamental active and fundamental reactive power of 8 channels. Plus, there are 2 kinds power on the overall system, the vector sum of power on the overall system and the absolute sum of power on the overall system. So there are 20 power data in total, as shown in Table 8-16. All registers are in the format of 32-bit 2' complement, readable and writable. All powers are updated once per 20ms, and are stabilized in 80ms.

**Table 8-16 Power Data Registers**

| No. | Address (Hex) | Register | Description | |
|---|---|---|---|---|
| 1 | 0x2C01 | PA | Phase A | Active power |
| 2 | 0x2C02 | PB | Phase B | |
| 3 | 0x2C03 | PC | Phase C | |
| 4 | 0x2C04 | QA | Phase A | Reactive power |
| 5 | 0x2C05 | QB | Phase B | |
| 6 | 0x2C06 | QC | Phase C | |
| 7 | 0x2C07 | PVEC | Vector sum of active power on the overall system | |
| 8 | 0x2C08 | PABS | Absolute sum of active power on the overall system | |
| 9 | 0x2C0C | QVEC | Vector sum of reactive power on the overall system | |
| 10 | 0x2C0D | QABS | Absolute sum reactive power on the overall system | |

| | | | | | |
|---|---|---|---|---|---|
| 11 | 0x2C14 | PFDA | Phase A | Fundamental active power | The gain of fundamental is 1.1834. When there is only fundament component and the power gain registers of every phase are set to 0, the ratio of fundamental active and reactive power to the total (fundamental and harmonic) active and reactive power is 1.1834. |
| 12 | 0x2C15 | PFDB | Phase B | | |
| 13 | 0x2C16 | PFDC | Phase C | | |
| 14 | 0x2C17 | QFDA | Phase A | Fundamental reactive power | |
| 15 | 0x2C18 | QFDB | Phase B | | |
| 16 | 0x2C19 | QFDC | Phase C | | |
| 17 | 0x2C1A | PFDVEC | Vector sum of fundamental active power on the overall system | | |
| 18 | 0x2C1B | PFDABS | Absolute sum of fundamental active power on the overall system | | |
| 19 | 0x2C1F | QFDVEC | Vector sum of fundamental reactive power on the overall system | | |
| 20 | 0x2C20 | QFDABS | Absolute sum of fundamental reactive power on the overall system | | |

**Table 8-17 Power Gain Registers**

| No. | Address | Register | Description | |
|---|---|---|---|---|
| 1 | 0x2A9D | SCPA | Active power gain of Phase A* | All registers are in the format of 32-bit 2' complement. Readable and writable. |
| 2 | 0x2AA1 | SCPB | Active power gain of Phase B | |
| 3 | 0x2AA5 | SCPC | Active power gain of Phase C | |
| 4 | 0x2AA9 | SCQA | Reactive power gain of Phase A | |
| 5 | 0x2AAA | SCQB | Reactive power gain of Phase B | |
| 6 | 0x2AAB | SCQC | Reactive power gain of Phase C | |
| 7 | 0x2AB4 | SCFDPA | Fundamental active power gain of Phase A | |
| 8 | 0x2AB5 | SCFDPB | Fundamental active power gain of Phase B | |
| 9 | 0x2AB6 | SCFDPC | Fundamental active power gain of Phase C | |
| 10 | 0x2AB7 | SCFDQA | Fundamental reactive power gain of Phase A | |
| 11 | 0x2AB8 | SCFDQB | Fundamental reactive power gain of Phase B | |
| 12 | 0x2AB9 | SCFDQC | Fundamental reactive power | |

| | | | | |
|---|---|---|---|---|
| | | | gain of Phase C | |

* Active power gain calibration is configured in 3 segments automatically.

## 8.10.2.Power Gain Calibration in Segments

To widen the dynamic range, the active power can be gain calibrated in segments. As shown in Figure 8-10, the chip is designed to gain calibrate the active power in 3 segments. Users can compare the calibrated current RMS to the preset current threshold in segment, and select the corresponding data register for active power gain calibration automatically.


**Figure 8-10 Gain Calibrating Active Power in 3 Segments**

The return difference is designed, the amplitude of which is decided by Return Difference Amplitude Control Register of Active Power Gain Calibration in Segments (MSKP, Table 8-18). This register is 5-bit register, located at bit16~bit20 of the register MTPARA0, sized from 0 to 31, which decides the return difference.

**Table 8-18 Return Difference Amplitude Control Register of Active Power Gain Calibration in Segments, MSKP**

| Register | Address | Width | Location | Description |
|---|---|---|---|---|
| MSKP | 0x2000 | 5 | Bit[20:16] of MTPARA0 | Return Difference Amplitude of Gain Calibration in Segments Control Register |

For example, if the register MSKP is set to 8, it means that the last 8 bits of the current threshold register for active power gain calibration determine the return difference. So, as shown in Figure 8-11, if the current threshold is set to 0xabcd (for example, the value of SCPTH0A), the higher return difference point of the threshold, THH, is 0xac00, and the lower return difference point of the threshold, THL, is 0xab00.

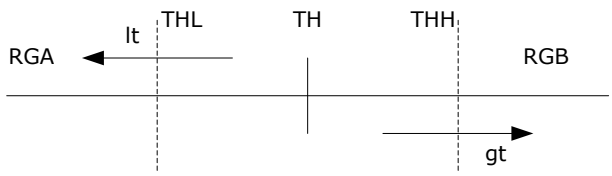



**Figure 8-11 Return Difference of the Gain Calibration**

Table 8-19 lists all active power gain calibration related registers. All registers are in the format of 32-bit 2’ complement (readable and writable).

**Table 8-19 Active Power Gain Calibration in Segments Related Registers**

| No. | Address | Register | Description | |
|---|---|---|---|---|
| 1 | 0x2A9D | SCPA** | Active power gain calibration of Phase A. The value of SCPAx (x=0~2) is written into this register. | |
| 2 | 0x2A9E | SCPA0 | Segment 0 | Active power gain calibration in segments of Phase A |
| 3 | 0x2A9F | SCPA1 | Segment 1 | |
| 4 | 0x2AA0 | SCPA2 | Segment 2 | |
| 5 | 0x2AA1 | SCPB** | Active power gain calibration of Phase B. The value of SCPBx (x=0~2) is written into this register. | |
| 6 | 0x2AA2 | SCPB0 | Segment 0 | Active power gain calibration in segments of Phase B |
| 7 | 0x2AA3 | SCPB1 | Segment 1 | |
| 8 | 0x2AA4 | SCPB2 | Segment 2 | |
| 9 | 0x2AA5 | SCPC** | Active power gain calibration of Phase C. The value of SCPCx (x=0~2) is written into this register. | |
| 10 | 0x2AA6 | SCPC0 | Segment 0 | Active power gain calibration in segments of Phase C |
| 11 | 0x2AA7 | SCPC1 | Segment 1 | |
| 12 | 0x2AA8 | SCPC2 | Segment 2 | |
| 13 | 0x2ACC | SCPTH0A* | Current threshold 0 for active power gain calibration in segments of Phase A | |
| 14 | 0x2ACD | SCPTH0B | Current threshold 0 for active power gain calibration in segments of Phase B | |
| 15 | 0x2ACE | SCPTH0C | Current threshold 0 for active power gain calibration in segments of Phase C | |
| 16 | 0x2ACF | SCPTH1A* | Current threshold 1 for active power gain calibration in segments of Phase A | |
| 17 | 0x2AD0 | SCPTH1B | Current threshold 1 for active power gain calibration in segments of Phase B | |
| 18 | 0x2AD1 | SCPTH1C | Current threshold 1 for active power gain calibration in segments of Phase C | |

* The calibrated current RMS (IRMS) is used to determine the segment to be used. Take Phase A for example, when IRMS<SCPTH0A, use the register SCPA0; when SCPTH0A<IRMS<SCPTH1A, use the register SCPA1; when IRMS>SCPTH1A, use the register SCPA2.

**When initializing the chip, users should write the value of the most-commonly-used register, such as SCPA0, SCPA1, or SCPA2, into the active power gain calibration register, such as SCPA.

Use Gain Calibration in Segments Control Register (PECEN, Table 8-20) to enable or disable gain calibration in segments. When gain calibration in segments is disabled, the value of Segment 0 register of active power gain calibration, such as SCPA0, is automatically written into the active power gain calibration register, such as SCPA.

**Table 8-20 Gain Calibration in Segments Control Register, PECEN**

| Register | Address | Width | Location | Description |
|---|---|---|---|---|
| PECEN | 0x2002 | 1 | Bit9 of MTPARA2 | Gain Calibration in Segments Control Register.<br>1, to disable; 0, to enable. |

**Table 8-21 Calibration in Segments and No-load Detection Checkup Register (LPAEC, 0x200E)**

| Register | Byte | Width (Bit) | Location | Description |
|---|---|---|---|---|
| LPAEC<br>0x200E | 0 | 8 | [7:0] | To indicate the segment for phase compensation and active power gain calibration of Phase A.<br>Bit0~Bit2, phase compensation.<br>000, Segment 0; 001, Segment 1; 010, Segment 2;<br>011, Segment 3; 100, Segment 4.<br>Bit4~bit5, active power gain calibration.<br>00, Segment 0; 01, Segment 1; 10, Segment 2. |
| | 1 | 8 | [15:8] | To indicate the segment for phase compensation and active power gain calibration of Phase B.<br>Bit8~Bit10, phase compensation.<br>000, Segment 0; 001, Segment 1; 010, Segment 2;<br>011, Segment 3; 100, Segment 4.<br>Bit12~Bit13, active power gain calibration.<br>00, Segment 0; 01, Segment 1; 10, Segment 2. |
| | 2 | 8 | [23:16] | To indicate the segment for phase compensation and active power gain calibration of Phase C.<br>Bit16~Bit18, phase compensation. |

| | | | | |
|---|---|---|---|---|
| | | | | 000, Segment 0; 001, Segment 1; 010, Segment 2;<br>011, Segment 3; 100, Segment 4.<br>Bit20~Bit21, active power gain calibration.<br>00, Segment 0; 01, Segment 1; 10, Segment 2. |
| | 3 | 8 | [31:24] | Bit24, to indicate the no-load status.<br>1, creeping; 0, starting |

## 8.10.3.Apparent Power/Power Factor Angle

As for V9003/V9103, when active and reactive power has the same gain, the relationship of apparent power (S), active power (P) and reactive power (Q) is as follows:

$$S = 1.647 \times \sqrt{P^2 + Q^2} \quad \text{or} \quad S = 1.647 \times \frac{P}{\cos\theta}$$

Where,

S: Apparent power, VA;

P: Active power, W;

Q: Reactive power, Var;

$\theta$: Power factor angle, as shown in Figure 8-8;

1.647 is the gain of apparent power.

Table 8-22 lists all apparent power registers and power factor angle data registers. All of them are 32-bit 2' complement registers, readable and writable. All apparent powers and power factor angles, both total and fundamental, are updated once per second, and are stabilized in 2 seconds.

**Table 8-22 Apparent Power Registers and Power Factor Angle Data Registers**

| No. | Address | Register | Description |
|---|---|---|---|
| 1 | 0x2C95 | APTA | Apparent power of Phase A |
| 2 | 0x2C96 | APTB | Apparent power of Phase B |
| 3 | 0x2C97 | APTC | Apparent power of Phase C |
| 4 | 0x2C98 | APTVEC | Vector sum of apparent power on the overall system |
| 5 | 0x2C99 | APTFDA | Fundamental apparent power of Phase A |
| 6 | 0x2C9A | APTFDB | Fundamental apparent power of Phase B |

| | | | |
|---|---|---|---|
| 7 | 0x2C9B | APTFDC | Fundamental apparent power of Phase C |
| 8 | 0x2C9C | APTFDVEC | Vector sum of fundamental apparent power on the overall system |
| 9 | 0x2C9D | PFAA | Power factor angle of Phase A |
| 10 | 0x2C9E | PFAB | Power factor angle of Phase B |
| 11 | 0x2C9F | PFAC | Power factor angle of Phase C |
| 12 | 0x2CA0 | PFAVEC | Vector sum of power factor angle on the overall system |
| 13 | 0x2CA1 | PFAFDA | Fundamental power factor angle of Phase A |
| 14 | 0x2CA2 | PFAFDB | Fundamental power factor angle of Phase B |
| 15 | 0x2CA3 | PFAFDC | Fundamental power factor angle of Phase C |
| 16 | 0x2CA4 | PFAFDVEC | Vector sum of fundamental power factor angle on the overall system |

## 8.11. Metering Energy

The power is accumulated into the positive and negative energy tanks according to the sign of the value, which is done once per 512 clock periods. When the energy accumulated in the tanks is more than the threshold, a pulse is generated, the corresponding pulse counter increments by 1, and the energy tank is subtracted by the threshold. When the pulse counter increments by 2, a CF pulse is output.

There are 20 power values for V9003/V9103, which need 36 energy tanks and pulse counters (Table 8-23). There is no negative energy tank for the absolute sum of power on the overall system. An energy tank is 64-bit 2’ complement, sharing 2 addresses. The pulse counters are 32-bit 2’ complement registers, sharing one address. All energy tanks and pulse counters are readable and writable.

All data in energy tanks or pulse counters, both total and fundamental, are updated in 78μs.

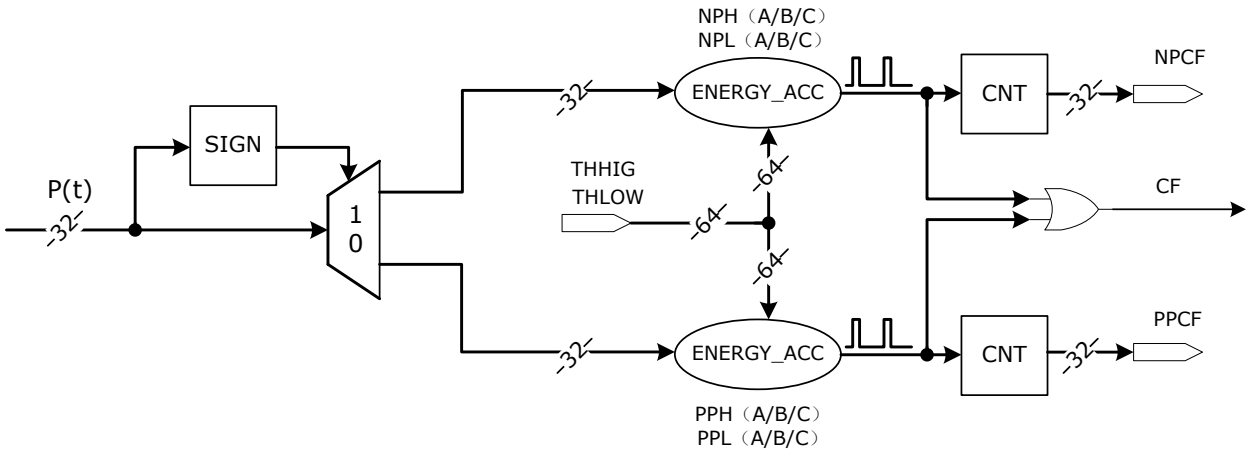



**Figure 8-12 Accumulating Power into the Energy Tanks**

**Table 8-23 Energy Tanks and CF Pulse Counters**

| No. | Address | Register | Description | |
|---|---|---|---|---|
| 1 | 0x2C09 | PABSH | Higher 32-bit of energy tank | Absolute sum of active energy tank and CF pulse counter on the overall system |
| 2 | 0x2C0A | PABSL | Lower 32-bit of energy tank | |
| 3 | 0x2C0B | PABSCF | CF pulse counter | |
| 4 | 0x2C0E | QABSH | Higher 32-bit of energy tank | Absolute sum of reactive energy tank and CF pulse counter on the overall system |
| 5 | 0x2C0F | QABSL | Lower 32-bit of energy tank | |
| 6 | 0x2C10 | QABSCF | CF pulse counter | |
| 7 | 0x2C1C | PFDABSH | Higher 32-bit of energy tank | Absolute sum of fundamental active energy tank and CF pulse counter on the overall system |
| 8 | 0x2C1D | PFDABSL | Lower 32-bit of energy tank | |
| 9 | 0x2C1E | PFDABSCF | CF pulse counter | |
| 10 | 0x2C21 | QFDABSH | Higher 32-bit of energy tank | Absolute sum of fundamental reactive energy tank and CF pulse counter on the overall system |
| 11 | 0x2C22 | QFDABSL | Lower 32-bit of energy tank | |

| | | | | |
|---|---|---|---|---|
| 12 | 0x2C23 | QFDABSCF | CF pulse counter | |
| 13 | 0x2C35 | NPAH | Higher 32-bit of energy tank | Negative active energy tank and CF pulse counter of Phase A |
| 14 | 0x2C36 | NPAL | Lower 32-bit of energy tank | |
| 15 | 0x2C37 | NPACF | CF pulse counter | |
| 16 | 0x2C38 | NPBH | Higher 32-bit of energy tank | Negative active energy tank and CF pulse counter of Phase B |
| 17 | 0x2C39 | NPBL | Lower 32-bit of energy tank | |
| 18 | 0x2C3A | NPBCF | CF pulse counter | |
| 19 | 0x2C3B | NPCH | Higher 32-bit of energy tank | Negative active energy tank and CF pulse counter of Phase C |
| 20 | 0x2C3C | NPCL | Lower 32-bit of energy tank | |
| 21 | 0x2C3D | NPCCF | CF pulse counter | |
| 22 | 0x2C3E | NQAH | Higher 32-bit of energy tank | Negative reactive energy tank and CF pulse counter of Phase A |
| 23 | 0x2C3F | NQAL | Lower 32-bit of energy tank | |
| 24 | 0x2C40 | NQACF | CF pulse counter | |
| 25 | 0x2C41 | NQBH | Higher 32-bit of energy tank | Negative reactive energy tank and CF pulse counter of Phase B |
| 26 | 0x2C42 | NQBL | Lower 32-bit of energy tank | |
| 27 | 0x2C43 | NQBCF | CF pulse counter | |
| 28 | 0x2C44 | NQCH | Higher 32-bit of energy tank | Negative reactive energy tank and CF pulse counter of Phase C |
| 29 | 0x2C45 | NQCL | Lower 32-bit of energy tank | |
| 30 | 0x2C46 | NQCCF | CF pulse counter | |
| 31 | 0x2C47 | NPVECH | Higher 32-bit of | Vector sum of negative active energy tank and |

| | | | | |
|---|---|---|---|---|
| | | | energy tank | CF pulse counter on the overall system |
| 32 | 0x2C48 | NPVECL | Lower 32-bit of energy tank | |
| 33 | 0x2C49 | NPVECCF | CF pulse counter | |
| 34 | 0x2C4A | NQVECH | Higher 32-bit of energy tank | Vector sum of negative reactive energy tank and CF pulse counter on the overall system |
| 35 | 0x2C4B | NQVECL | Lower 32-bit of energy tank | |
| 36 | 0x2C4C | NQVECCF | CF pulse counter | |
| 37 | 0x2C4D | NPFDAH | Higher 32-bit of energy tank | Fundamental negative active energy tank and CF pulse counter of Phase A |
| 38 | 0x2C4E | NPFDAL | Lower 32-bit of energy tank | |
| 39 | 0x2C4F | NPFDACF | CF pulse counter | |
| 40 | 0x2C50 | NPFDBH | Higher 32-bit of energy tank | Fundamental negative active energy tank and CF pulse counter of Phase B |
| 41 | 0x2C51 | NPFDBL | Lower 32-bit of energy tank | |
| 42 | 0x2C52 | NPFDBCF | CF pulse counter | |
| 43 | 0x2C53 | NPFDCH | Higher 32-bit of energy tank | Fundamental negative active energy tank and CF pulse counter of Phase C |
| 44 | 0x2C54 | NPFDCL | Lower 32-bit of energy tank | |
| 45 | 0x2C55 | NPFDCCF | CF pulse counter | |
| 46 | 0x2C56 | NQFDAH | Higher 32-bit of energy tank | Fundamental negative reactive energy tank and CF pulse counter of Phase A |
| 47 | 0x2C57 | NQFDAL | Lower 32-bit of energy tank | |
| 48 | 0x2C58 | NQFDACF | CF pulse counter | |
| 49 | 0x2C59 | NQFDBH | Higher 32-bit of energy tank | Fundamental negative reactive energy tank and CF pulse counter of Phase B |
| 50 | 0x2C5A | NQFDBL | Lower 32-bit of energy tank | |

| | | | | |
|---|---|---|---|---|
| 51 | 0x2C5B | NQFDBCF | CF pulse counter | |
| 52 | 0x2C5C | NQFDCH | Higher 32-bit of energy tank | Fundamental negative reactive energy tank and CF pulse counter of Phase C |
| 53 | 0x2C5D | NQFDCL | Lower 32-bit of energy tank | |
| 54 | 0x2C5E | NQFDCCF | CF pulse counter | |
| 55 | 0x2C5F | NPFDVECH | Higher 32-bit of energy tank | Vector sum of fundamental positive energy tank and CF pulse counter on the overall system |
| 56 | 0x2C60 | NPFDVECL | Lower 32-bit of energy tank | |
| 57 | 0x2C61 | NPFDVECCF | CF pulse counter | |
| 58 | 0x2C62 | NQFDVECH | Higher 32-bit of energy tank | Vector sum of fundamental positive reactive energy tank and CF pulse counter on the overall system |
| 59 | 0x2C63 | NQFDVECL | Lower 32-bit of energy tank | |
| 60 | 0x2C64 | NQFDVECCF | CF pulse counter | |
| 61 | 0x2C65 | PPAH | Higher 32-bit of energy tank | Positive active energy tank and CF pulse counter of Phase A |
| 62 | 0x2C66 | PPAL | Lower 32-bit of energy tank | |
| 63 | 0x2C67 | PPACF | CF pulse counter | |
| 64 | 0x2C68 | PPBH | Higher 32-bit of energy tank | Positive active energy tank and CF pulse counter of Phase B |
| 65 | 0x2C69 | PPBL | Lower 32-bit of energy tank | |
| 66 | 0x2C6A | PPBCF | CF pulse counter | |
| 67 | 0x2C6B | PPCH | Higher 32-bit of energy tank | Positive active energy tank and CF pulse counter of Phase C |
| 68 | 0x2C6C | PPCL | Lower 32-bit of energy tank | |
| 69 | 0x2C6D | PPCCF | CF pulse counter | |
| 70 | 0x2C6E | PQAH | Higher 32-bit of | Positive reactive energy tank and CF pulse |

| | | | | |
|---|---|---|---|---|
| | | | energy tank | counter of Phase A |
| 71 | 0x2C6F | PQAL | Lower 32-bit of energy tank | |
| 72 | 0x2C70 | PQACF | CF pulse counter | |
| 73 | 0x2C71 | PQBH | Higher 32-bit of energy tank | Positive reactive energy tank and CF pulse counter of Phase B |
| 74 | 0x2C72 | PQBL | Lower 32-bit of energy tank | |
| 75 | 0x2C73 | PQBCF | CF pulse counter | |
| 76 | 0x2C74 | PQCH | Higher 32-bit of energy tank | Positive reactive energy tank and CF pulse counter of Phase C |
| 77 | 0x2C75 | PQCL | Lower 32-bit of energy tank | |
| 78 | 0x2C76 | PQCCF | CF pulse counter | |
| 79 | 0x2C77 | PPVECH | Higher 32-bit of energy tank | Vector sum of active energy tank and CF pulse counter on the overall system |
| 80 | 0x2C78 | PPVECL | Lower 32-bit of energy tank | |
| 81 | 0x2C79 | PPVECCF | CF pulse counter | |
| 82 | 0x2C7A | PQVECH | Higher 32-bit of energy tank | Vector sum of reactive energy tank and CF pulse counter on the overall system |
| 83 | 0x2C7B | PQVECL | Lower 32-bit of energy tank | |
| 84 | 0x2C7C | PQVECCF | CF pulse counter | |
| 85 | 0x2C7D | PPFDAH | Higher 32-bit of energy tank | Fundamental positive active energy tank and CF pulse counter of Phase A |
| 86 | 0x2C7E | PPFDAL | Lower 32-bit of energy tank | |
| 87 | 0x2C7F | PPFDACF | CF pulse counter | |
| 88 | 0x2C80 | PPFDBH | Higher 32-bit of energy tank | Fundamental positive active energy tank and CF pulse counter of Phase B |
| 89 | 0x2C81 | PPFDBL | Lower 32-bit of energy tank | |

| | | | | |
|---|---|---|---|---|
| 90 | 0x2C82 | PPFDBCF | CF pulse counter | |
| 91 | 0x2C83 | PPFDCH | Higher 32-bit of energy tank | Fundamental positive active energy tank and CF pulse counter of Phase C |
| 92 | 0x2C84 | PPFDCL | Lower 32-bit of energy tank | |
| 93 | 0x2C85 | PPFDCCF | CF pulse counter | |
| 94 | 0x2C86 | PQFDAH | Higher 32-bit of energy tank | Fundamental positive reactive energy tank and CF pulse counter of Phase A |
| 95 | 0x2C87 | PQFDAL | Lower 32-bit of energy tank | |
| 96 | 0x2C88 | PQFDACF | CF pulse counter | |
| 97 | 0x2C89 | PQFDBH | Higher 32-bit of energy tank | Fundamental positive reactive energy tank and CF pulse counter of Phase B |
| 98 | 0x2C8A | PQFDBL | Lower 32-bit of energy tank | |
| 99 | 0x2C8B | PQFDBCF | CF pulse counter | |
| 100 | 0x2C8C | PQFDCH | Higher 32-bit of energy tank | Fundamental positive reactive energy tank and CF pulse counter of Phase C |
| 101 | 0x2C8D | PQFDCL | Lower 32-bit of energy tank | |
| 102 | 0x2C8E | PQFDCCF | CF pulse counter | |
| 103 | 0x2C8F | PPFDVECH | Higher 32-bit of energy tank | Vector sum of fundamental active energy tank and CF pulse counter on the overall system |
| 104 | 0x2C90 | PPFDVECL | Lower 32-bit of energy tank | |
| 105 | 0x2C91 | PPFDVECCF | CF pulse counter | |
| 106 | 0x2C92 | PQFDVECH | Higher 32-bit of energy tank | Vector sum of fundamental reactive energy tank and CF pulse counter on the overall system |
| 107 | 0x2C93 | PQFDVECL | Lower 32-bit of energy tank | |
| 108 | 0x2C94 | PQFDVECCF | CF pulse counter | |

# 8.12. No-load Detection

V9003/V9103 include a no-load threshold feature on the active energy that eliminates any creep effects in the meter. It works as follows. The no-load detection threshold power, the value of the register THCRP, is accumulated into the no-load detection energy tank, and other power is accumulated into the corresponding energy tank simultaneously, for example, accumulating the active power of Phase A into the positive or negative active energy tank of Phase A. If the value of the no-load detection energy tank amounts to the energy threshold, for example, THHIG, THLOW, and so on, firstly, other energy tanks are cleared, and the system enters creeping. If the values of other energy tanks amount to the energy threshold firstly, the no-load detection energy tank is cleared, and the system enters starting.

In practice, users can control no-load detection by software, see "Calibration Examples" for details.

Users can configure the Anti-creeping Control Register (CRPEN, Table 8-25) to enable or disable anti-creeping.

**Table 8-24 Energy Threshold Registers and No-load Detection Threshold Register**

<table>
<tr><th>No.</th><th>Address</th><th>Register</th><th colspan="2">Description</th></tr>
<tr><td>1</td><td>0x2CA6</td><td>THHIG</td><td rowspan="2">Total energy (fundamental and harmonic) threshold</td><td rowspan="4">Energy threshold register, 64-bit 2’ complement, sharing two addresses each. Readable and writable.</td></tr>
<tr><td>2</td><td>0x2CA7</td><td>THLOW</td></tr>
<tr><td>3</td><td>0x2CA8</td><td>THFDH</td><td rowspan="2">Fundamental energy threshold</td></tr>
<tr><td>4</td><td>0x2CA9</td><td>THFDL</td></tr>
<tr><td>5</td><td>0x2CAA</td><td>THCRP</td><td colspan="2">No-load detection threshold power. In the format of 32-bit 2’ complement, readable and writable.</td></tr>
</table>

**Table 8-25 Anti-creeping Control Register, CRPEN**

| Register | Address | Width | Location | Description |
|---|---|---|---|---|
| CRPEN | 0x2002 | 1 | Bit10 of MTPARA2 | Anti-creeping Control Register.<br>1, to disable; 0, to enable. |

# 8.13. Calculating Voltage and Current RMS

Take Phase A as an example to show how to calculate the voltage and current RMS, as shown in Figure 8-13. There are 14 RMS values. Table 8-26 lists all current and voltage RMS related registers. All of them are 32-bit 2’ complement registers (readable and writable).

All data in the RMS registers are updated once per 20ms, and are stabilized in 80ms.

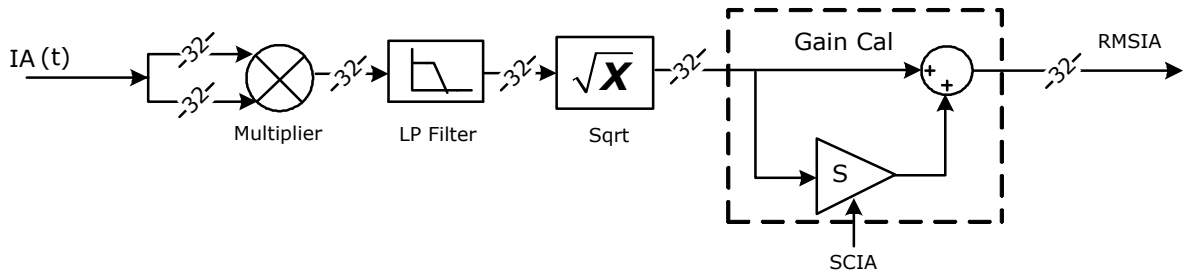



**Figure 8-13 Calculating the Voltage and Current RMS in Phase A**

**Table 8-26 Voltage/Current RMS and RMS Gain Registers**

| No. | Address | Register | Description | |
|---|---|---|---|---|
| 1 | 0x2A4F | RMSUA | Voltage RMS of Phase A | |
| 2 | 0x2A50 | RMSIA | Current RMS of Phase A | |
| 3 | 0x2A51 | RMSUB | Voltage RMS of Phase B | |
| 4 | 0x2A52 | RMSIB | Current RMS of Phase B | |
| 5 | 0x2A53 | RMSUC | Voltage RMS of Phase C | |
| 6 | 0x2A54 | RMSIC | Current RMS of Phase C | |
| 7 | 0x2A55 | RMSIN | Current RMS of Channel IN | |
| 8 | 0x2A56 | RMSM | Voltage RMS of M Channel | |
| 9 | 0x2A87 | RMSFDUA | Fundamental Voltage RMS of Phase A | The gain of fundamental voltage/current RMS is 1.0878. When there is only fundament component, and the RMS gain calibration registers of every phase are set to 0, the ratio of fundamental RMS to the total RMS (fundamental and harmonic) is 1.0878. |
| 10 | 0x2A88 | RMSFDIA | Fundamental Current RMS of Phase A | |
| 11 | 0x2A89 | RMSFDUB | Fundamental Voltage RMS of Phase B | |
| 12 | 0x2A8A | RMSFDIB | Fundamental Current RMS of Phase B | |
| 13 | 0x2A8B | RMSFDUC | Fundamental Voltage RMS of Phase C | |
| 14 | 0x2A8C | RMSFDIC | Fundamental Current RMS of Phase C | |
| 15 | 0x2AAC | SCUA | Voltage RMS calibration of Phase A | |

| | | | |
|---|---|---|---|
| 16 | 0x2AAD | SCIA | Current RMS calibration of Phase A |
| 17 | 0x2AAE | SCUB | Voltage RMS calibration of Phase B |
| 18 | 0x2AAF | SCIB | Current RMS calibration of Phase B |
| 19 | 0x2AB0 | SCUC | Voltage RMS calibration of Phase C |
| 20 | 0x2AB1 | SCIC | Current RMS calibration of Phase C |
| 21 | 0x2AB2 | SCIN | Current RMS calibration of Channel IN |
| 22 | 0x2AB3 | SCM | Voltage RMS calibration of M Channel |
| 23 | 0x2ABA | SCFDUA | Fundamental Voltage RMS calibration of Phase A |
| 24 | 0x2ABB | SCFDIA | Fundamental Current RMS calibration of Phase A |
| 25 | 0x2ABC | SCFDUB | Fundamental Voltage RMS calibration of Phase B |
| 26 | 0x2ABD | SCFDIB | Fundamental Current RMS calibration of Phase B |
| 27 | 0x2ABE | SCFDUC | Fundamental Voltage RMS calibration of Phase C |
| 28 | 0x2ABF | SCFDIC | Fundamental Current RMS calibration of Phase C |

# 8.14. Generating Pulse and CF Interrupt

The energy metering DSP of V9003/V9103 can generate two CF interrupts, CF1 and CF2.

When working normally, the sampling frequency of ADCs is configured to 819.2kHz, and the output pulse width is 80ms. When the output frequency of the CF pulse is higher than 5.56Hz, namely the period of the output pulse is less than 160ms, the duty cycle should be 50%.

In the metering configuration register MTPARA2, there are two registers, CFON and CFSEL (Table 8-27), to turn on/off CF1/CF2, and to select the source of CF pulse.

**Table 8-27 CF Pulse Control Registers, CFON and CFSEL**

| Register | Width | Location | Description |
|---|---|---|---|
| CFON | 2 | Bit[1:0] of MTPARA2 | Bit1, to turn on or off CF2;<br>Bit0, to turn on or off CF1.<br>1, on; 0, off. |
| CFSEL | 4 | Bit[7:4] of MTPARA2 | To select the source of CF1/CF2.<br>0, to generate CF pulses based on the vector sum of active and reactive energy on the overall system; |

| | | | |
|---|---|---|---|
| | | | 1, to generate CF pulses based on the absolute sum of active and reactive energy on the overall system;<br>2, to generate CF pulses based on the active and reactive energy of Phase A;<br>3, to generate CF pulses based on the active and reactive energy of Phase B;<br>4, to generate CF pulses based on the active and reactive energy of Phase C;<br>5, to generate CF pulses based on the vector sum of fundamental active and reactive energy on the overall system;<br>6, to generate CF pulses based on the absolute sum of fundamental active and reactive energy on the overall system;<br>7, to generate CF pulses based on the fundamental active and reactive energy of Phase A;<br>8, to generate CF pulses based on the fundamental active and reactive energy of Phase B;<br>9, to generate CF pulses based on the fundamental active and reactive energy of Phase C. |

# 8.15. Frequency-phase Measurement

In V9003/V9103, the energy metering DSP measures the line frequency of the input signal via counting the number of the time intervals between two zero-crossings in 64 sine waves. As for a 50Hz input signal, when $f_{MCU}$=6.5536MHz, there are 256 time intervals in a wave period, so, the frequency value in the frequency-phase registers is 0x4000 (in decimal, 16384).

The line frequency of the input signal can be calculated using the following equation:

Line Frequency=$f_{ADC}$ / Frequency value of the frequency-phase registers (in decimal)

Where, $f_{ADC}$ is the sampling frequency of ADCs, Hz.

In V9003/V9103, the energy metering DSP measures the relative phase between the current and voltage channels when a zero crossing is detected. The phase difference is from 0 to 255, corresponding to the phase angle from 0 degree to 360 degrees.

Both frequency and phase data of every signal are stored in the 32-bit 2' complement register (readable and writable, Table 8-28), of which lower 16 bits give the phase measurement, and

higher 16 bits give the frequency measurement.

The data of higher 16-bit of the registers are updated once per 1.28 seconds and are stabilized in 1.28 seconds; and the data of the lower 16-bit of the registers are updated once per 20ms and are stabilized in 40ms.

When reading the frequency-phase data registers, please note that:

1. Read the register FPUA first, to obtain the phase of Voltage Channel UA at the moment T, and then, read other registers to obtain the phases of other channels at the moment T;
2. If other registers are accessed directly without accessing the register FPUA in the next time, the lower 16 bits of these registers holds the previous phases.
3. If other registers are accessed directly without accessing the register FPUA after reset, the lower 16 bits of these registers gives zeros.

**Table 8-28 Frequency-Phase Data Registers**

| No. | Address | Register | Phase | Description |
|---|---|---|---|---|
| 1 | 0x2008 | FPUA | Voltage Channel UA | Higher 16-bit, frequency measurement<br>Lower 16-bit, phase measurement, only bit7~bit0 are active. |
| 2 | 0x2009 | FPUB | Voltage Channel UB | |
| 3 | 0x200A | FPUC | Voltage Channel UC | |
| 4 | 0x200B | FPIA | Current Channel IA | |
| 5 | 0x200C | FPIB | Current Channel IB | |
| 6 | 0x200D | FPIC | Current Channel IC | |

# 8.16. Using the M Channel

## 8.16.1.The M Channel

The M Channel can be configured to measure Ground, temperature, RTC battery voltage, main battery voltage and external voltage sources. Figure 8-14 shows the schematics of M Channel.

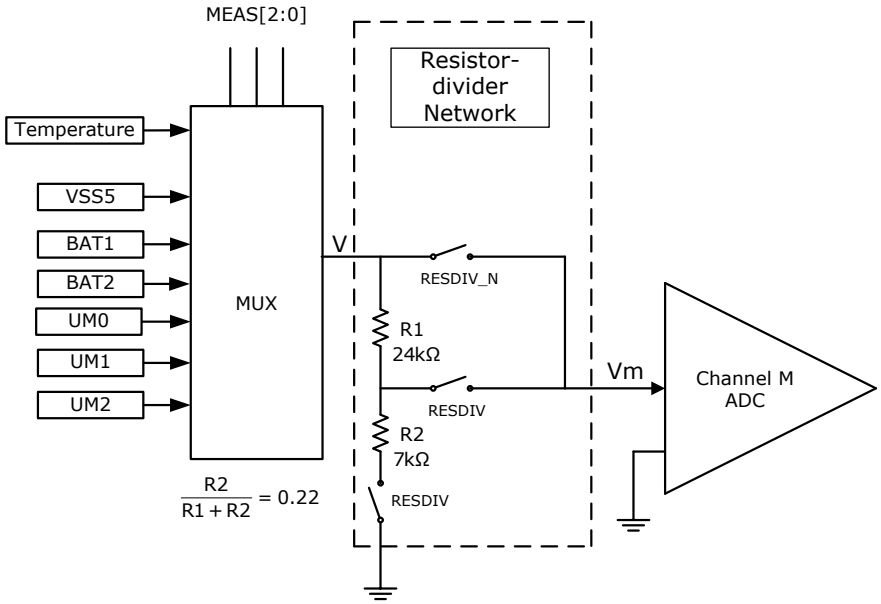



**Figure 8-14 Schematics of M Channel**

There is only one ADC for M Channel, so, battery voltage, temperature and external voltage sources should be measured alternatively, as shown in Figure 8-14. As for V9003/V9103, the pins BAT1, UM0, UM1, and UM2 all can be configured to measure main battery voltage and external voltage sources, and BAT2 can be used to measure RTC battery voltage only.

Configure the bits MEAS<2:0> of the register FRQCFG0 (0x1B15) to select the signal to be measured (Table 8-29).

**Table 8-29 Bit Description of MEAS<2:0>**

| MEAS<2:0> | Description |
|---|---|
| 000 | To measure Ground |
| 001 | To measure temperature |
| 010 | To measure main battery voltage or external voltage sources via BAT1 |
| 011 | To measure RTC battery voltage via BAT2 |
| 100 | N/A |
| 101 | To measure main battery voltage or external voltage sources via UM0* |
| 110 | To measure main battery voltage or external voltage sources via UM1* |
| 111 | To measure main battery voltage or external voltage sources via UM2* |

* UM0~UM2 share the pins (P8.5~P8.7) with SEG37~SEG39. When these pins are used to measure external voltage sources, they should be configured as input/output disabled, and the

corresponding LCD drivers should be turned off.

The M Channel can be used to measure the DC voltage and voltage RMS of the input signals. Both of them are updated once per 20ms, and are stabilized in 80ms.

**Table 8-30 Related Data Registers to M Channel**

| No. | Address | Register | Description | |
|---|---|---|---|---|
| 1 | 0x2819 | DCM | DC component of the voltage in M Channel, updated once per 20ms | 32-bit 2' complement, readable and writable |
| 2 | 0x2A56 | RMSM | Voltage RMS in M Channel, updated once per 20ms | |

## 8.16.2.Measuring Temperature

When MEAS[2:0]=000 or 001, M Channel is configured to measure Ground or temperature. In this application, the bit RESDIV (Bit7 of ICtrl1, 0x1B0D) is automatically set to 0, meaning that the resistor-divider network is turned off. The data can be read out of the register DCM.

M Channel can measure the temperature over the range of -40~+85°C. The temperature accuracy is 1°C, and the data of the register DCM is updated once per 20ms, and it is stabilized in 80ms.

**Table 8-31 Bias Current Control Register 1 of Current Channels (ICtrl1, 0x1B0D)**

| 0x1B0D, R/W | Bias Current Control Register 1 of Current Channels, ICtrl1 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | Bit4 | bit3 | bit2 | bit1 | bit0 |
| | RESDIV | REFGITM | REFBITM | | ADIT2M <1> | ADIT2M <0> | ADIT1M <1> | ADIT1M <0> |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 8-32 Bit Description of ICtrl1**

| Bit | | Function | Description |
|---|---|---|---|
| Bit[1:0] | ADIT1M<1:0> | To adjust the biasing current by the 1st modulator of M Channel | 00, ×1; 01, ×1.5; 10, ×2; 11, ×2.5 |
| Bit[3:2] | ADIT2M<1:0> | To adjust the biasing current by the 2nd modulator of M Channel | 00, ×1; 01, ×1.5; 10, ×2; 11, ×2.5 |
| Bit5 | REFBITM | To adjust the reference buffer biasing current by | 0, ×1; 1, ×1.33 |

| | | | |
|---|---|---|---|
| | | M Channel ADC | |
| Bit6 | REFGITM | To adjust the reference generation biasing current by M Channel ADC | 0, ×1; 1, ×1.5 |
| Bit7 | RESDIV | To enable resistor-divider network, the division ratio is 0.22. | 1, to enable. 0, by default. |

## 8.16.3.Measuring Battery Voltage and External Voltage

In V9003/V9103, the pins BAT1/UM0/UM1/UM2 have the same function. All of them can be configured for main battery and external voltage measurement. And, BAT2 can be used to measure the RTC battery only. See "Bit Description of MEAS<2:0>" for details.

As shown in Figure 8-14, the relationship between battery voltage or external voltage (V) and the input signal into the M Channel ADC (Vm) is determined by the bit RESDIV (bit 7 of ICtrl1).

When RESDIV=1, the resistor-divider network is enabled, and the relationship between V and Vm are as follows:

$$Vm = \frac{R2}{R1 + R2} \times V$$

of which, the value of V is over the range of -200mV ~ +4000mV. The power consumption of the resistor-divider network (P) is determined by the following formula:

$$P = \frac{V}{R1 + R2}$$

When RESDIV=0, the network is disabled, and the relationship between V and Vm is as follows:

$$Vm = V$$

of which, the value of V is over the range of -200mV ~+1.185V (the reference voltage).

The value of Vm can be read out of the registers DCM (0x2819) and RMSM (0x2A56).

Figure 16-6 and Figure 16-7 show the measurement results of M Channel when the resistor-divider network is enabled and disabled. When M Channel is used to measure main battery voltage and external voltage, the bit RESDIV is set to 0, by default.

# 8.17. Calibrating

## 8.17.1.Formulas for Related Registers

The following formulas can be used for both total (fundamental and harmonic) and fundamental energy metering.

**1. Related registers to current and voltage RMS**

All registers relative to current and voltage RMS, see "Calculating Voltage and Current RMS" for details.

The relationship between the value of these registers and the input signals follows the formula:

$$Value = V \times G \times K \qquad \text{Equation 8-1}$$

Where

V: the RMS value of input signal, mV;

G: the gain;

K: a constant. For total, $K=1.75\times10^6$; for fundamental, $K=1.9\times10^6$

For example,

In Channel IA, when the value of the sampled signal is 30.1mV, and the gain is 2, the value of the register RMSIA (0x2A50) should be:

$Value=30.1\times2\times1.75\times10^6=0x6478370$

**2. Power registers**

As for V9003/V9103, there are active, reactive, and apparent power values. So, there are 30 data registers related to power, see "Calculating Power" for details.

The value of the register can be calculated using the following formula:

$$Value = Vi \times Gi \times Vv \times Gv \times K \times \cos\theta \times p \qquad \text{Equation 8-2}$$

Where

Vi and Vv : the value of the input current and voltage, respectively;

Gi and Gv : the gains of the current and voltage channels, respectively;

$\cos\theta$ : the power factor;

P: the phase number;

K: the coefficient. For active and reactive power registers, K=858; for apparent power

registers, K=1414.

For example,

When the Vv is 9.3mV, Gv is 2, Vi is 30.1mV, Gi is 2, and cosθ is 1, the value of the register PABS (0x2C08) of 3-phase meter for 4-wire service should be

$$\text{Value} = 9.3 \times 2 \times 30.1 \times 2 \times 858 \times 3$$
$$= 0x2BFA6F$$

The value of the register APTVEC (0x2C98) of 3-phase meter for 4-wire service should be

$$\text{Value} = 9.3 \times 2 \times 30.1 \times 2 \times 1414 \times 3$$
$$= 0x4879E6$$

**3. Frequency-phase registers**

All frequency-phase registers, see "Frequency-phase Measurement" for details.

If the frequency value (higher 16 bits) of frequency-phase measurement register is set to "Freq" (in decimal), the actual line frequency should be:

$$Vhz = \frac{f_{ADC}}{Freq} \qquad \text{Equation 8-3}$$

Where, $f_{ADC}$ is the sampling frequency of ADCs, Hz.

**4. Power factor angle registers**

All power factor angle registers, see "Apparent Power/Power Factor Angle" for details.

If the value of the power factor angle register is set to "value", the actual power factor angle should be:

$$\theta = \frac{Value}{937010} \qquad \text{Equation 8-4}$$

The accuracy of the power factor angle is 0.1 degree.

**5. Energy threshold registers**

All energy threshold registers, see "No-load Detection" for details.

The values of energy threshold registers can be calculated with the following formula:

$$PGAT = P \times T \times 6400 \qquad \text{Equation 8-5}$$

Where,

P: the total Equation 8-5 or fundamental active power at the nominal current and voltage;

T: Time constant, can be calculated with the formula

$$T = \frac{3600 \times 1000}{PulseConstant \times U_n \times I_n \times PhaseNumber}$$

**6. No-load detection threshold register**

The no-load detection threshold register, see "No-load Detection" for details.

The value of this register can be calculated with the following formula:

$$PGAT = K \times P \qquad \text{Equation 8-6}$$

Where,

P: the active or apparent power at the nominal current and voltage;

K: the starting current. For a Level 1 meter, K≥4‰.

**Note:** The threshold for creeping must be less than that for starting. To ensure that a meter can work normally, K can be set to a half of the starting current. So, as for a Level 1 meter, K=2‰.

**7. Gain calibration registers**

All registers related to gain calibration: power gain calibration registers and RMS gain calibration registers, see "Power Gain Calibration in Segments" for details.

$$S = 2^{31}(\frac{1}{1+e} - 1) + S_1(\frac{1}{1+e}) \qquad \text{Equation 8-7}$$

Where,

S: the gain, in the form of complementary code;

$S_1$: the original gain;

e: the error.

**8. Phase compensation registers**

All phase compensation registers, see "Phase Compensation" for details.

$$N = \frac{3011}{2} \times Error \times \frac{f_{ADC}}{819200} \qquad \text{Equation 8-8}$$

Where, $f_{ADC}$ is the sampling frequency of ADCs, Hz.

The phase compensation value of Phase A is in bit7~bit0 of AECx (x=0~4), of Phase B is in bit15~bit8 of AECx (x=0~4), and, of Phase C is in bit 23~bit16 of AECx (x=0~4), of which, bit7/bit15/bit23 is the sign bit, which should be set to 1, meaning the compensation value is negative.

9. **Current Threshold registers for phase compensation and gain calibration in segments, see "**Phase Compensation**" and "**Power Gain Calibration in Segments**" for details**

$$Value = ValueI \times K \qquad \text{Equation 8-9}$$

Where,

ValueI: the data of the current RMS register at the nominal current, via;

K: the segment point. For example, 15%$I_b$ means K=0.15.

## 8.17.2.Calibrating Energy

**1. Coarse Calibration**

The energy can be calibrated coarsely via the energy threshold registers.

As for a meter, the magnitude of the sampled signal and the gain is constant, so the energy threshold values can be obtained via Equation 8-5. The active and reactive energy threshold is the same. For total energy metering, these registers are THHIG and THLOW; for fundamental energy metering, these registers are THFDH and THFDL.

**2. Fine Calibration**

1) Select the Phase A for metering, and set the power factor to 1.0. Power on the calibration equipment with 100%$I_b$ when gain calibration in segments is disabled; power on the calibration equipment with the current in the segment range when gain calibration in segments is enabled.

2) Read the energy errors, use Equation 8-7 to calculate the gain of Phase A and write the gain into the register SCPA.

3) Switch the power factor to 0.5L, and read the energy errors. Use Equation 8-8 to calculate the phase compensation value, and write it and the sign into the lower 8 bits of the registers AEC0~AEC4.

4) Use the same method to compensate the phase of Phase B and Phase C;

5) Switch the calibration equipment to outputting power on the overall system, to detect the error. If the error has not been calibrated, repeat the above steps.

## 8.17.3.Calibrating Current RMS

1. Select Phase A for energy metering;

2. Write 0 into the register SCIA (for total) or SCFDIA (for fundamental);
3. Power on the calibration equipment with 100%$I_n$ current;
4. Use Equation 8-1 to obtain the value of RMSIA, $I_0$.
5. Calibrate the value $I_0$ to make it equal to $I_n$, and obtain the constant current RMS ratio D via the formula $I_0 \times D = I_n$.
6. Read the RMS value $I_1$ (shown in the energy meter) of Channel IA, which is the product of the actual reading of the current RMS register and the current RMS ratio D.
7. Calculate the gain of Channel IA with the following formula:

   Firstly, calculate the error:

$$e = \frac{I_1 - I_n}{I_n} \qquad \text{Equation 8-10}$$

   Then, use Equation 8-7 to calculate the gain.
8. Use the same method to calibrate the current RMS gain when Channel IB or Channel IC is selected for energy metering.

Note: When the current powered on the energy meter is less than the starting current, the LCD of the energy meter cannot show the current RMS $I_1$.

## 8.17.4.Calibrating Voltage RMS

1. Select Phase A for energy metering;
2. Write 0 into the register SCUA (for total) or SCFDUA (for fundamental);
3. Power on the calibration equipment with 100%$U_n$ voltage;
4. Use Equation 8-1 to obtain the value of RMSUA, $U_0$.
5. Calibrate the value $U_0$ to make it equal to $U_n$, and obtain the constant voltage RMS ratio D via the formula $U_0 \times D = U_n$.
6. Read the RMS value $U_1$ (shown in the energy meter) of the voltage channel, which is the product of the actual reading of the voltage RMS register and the voltage RMS ratio D.
7. Calculate the gain of the voltage channel with the following formula:

   Firstly, calculate the error:

$$e = \frac{U_1 - U_n}{U_n} \qquad \text{Equation 8-11}$$

Then, use Equation 8-7 to calculate the gain.

8. Use the same method to calibrate the voltage RMS gain when Phase B or Phase C is selected for energy metering.

## 8.17.5.Calibration Flow

Figure 8-15 shows the calibration flow.

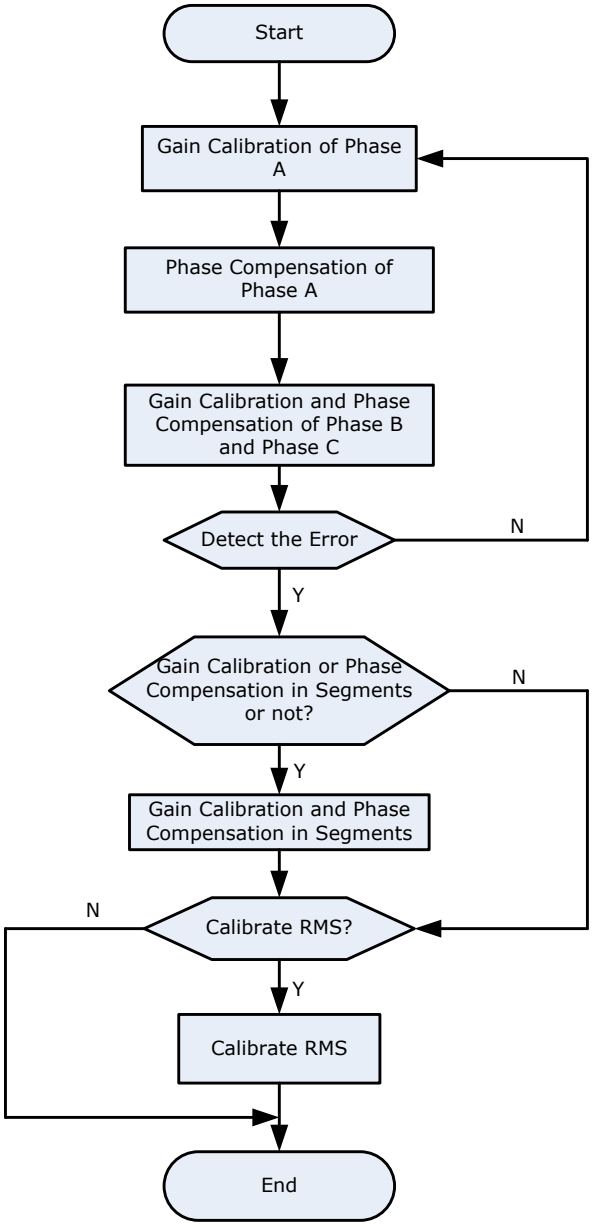



**Figure 8-15 Calibration Flow**

## 8.17.6.Calibration Examples

Take the total energy metering of a 3-phase 4-wire meter for example to instruct the calibration process.

### 8.17.6.1. Parameters of The Meter

Nominal current ($I_n$): 1.5A

Nominal voltage ($U_n$): 220V

Pulse constant: 6400

Accuracy Level: 1.0

### 8.17.6.2. Parameters for Design

The amplitude of sampled signal in current channels when powering on the calibration equipment with 100%$I_n$: 5mV

The amplitude of sampled signal in voltage channels when powering on the calibration equipment with 100%$I_n$: 10mV

### 8.17.6.3. Configuring Gains of ADCs

The gain of current channels: ×2

The gain of voltage channels: ×1

### 8.17.6.4. Calculating the Energy Threshold

1. To calculate the value of instantaneous power:

$$\text{Value} = Vi \times Gi \times Vv \times Gv \times K \times \cos\theta \times p$$
$$= 5 \times 2 \times 10 \times 1 \times 858 \times 1 \times 3 = 257400 = 0x3ED78$$

2. To calculate the energy threshold value:

$$\text{Value} = P \times T \times 6400$$
$$= \frac{6400 \times 257400 \times 3600 \times 1000}{6400 \times 1.5 \times 220 \times 3} = 936000000 = 0x37CA3A00$$

0x37CA3A00 is the threshold value, which can be used to configure the value of related registers:

THHIG (0x2CA6): 0x0

THLOW (0x2CA7): 0x37CA3A00

## 8.17.6.5. No-load Detection

In practice, users can perform no-load detection in V9003/V9103 by software.

For example, a meter, In=10A (60), Un=220V, Pulse constant = 1200, starting current =0.4%Ib, then, the time for staring a meter should be:

$$T = \frac{3600 \times 1000}{1200 \times 220 \times 10 \times 0.004} = 341s$$

If no pulse is generated in 341s, the system enters creeping. In creeping, write 1 into the register CLRACC (bit11 of MTPARA2, 0x2002) to clear the energy tanks, wait 1ms, and then, write 0 into CLRACC. Users can set the time for stating a meter to be 682s to prohibit the system from entering creeping prematurely. In practice, RTC second interrupt is used for timing.

## 8.17.6.6. Calculating The Ratio Factor of Current/Voltage RMS

Use Equation 8-1 to calculate the value of RMS related registers (Table 8-26). These values should be converted into actual RMS values of current using a ratio factor.

1. To calculate the ratio factor of current RMS in Channel IA.

   Use Equation 8-1 to calculate the value of the register RMSIA.

   $$Value = V \times G \times K$$
   $$= 0.005 \times 2 \times 1.75 \times 10^6 = 17500 = 0x445C$$

   If the actual value of current is 1.50A, the ratio factor D can be calculated with the following formula:

   $$Ivalue = Value \times D$$
   $$150 = 17500 \times D$$
   $$D = \frac{150}{17500} = 0.0085714$$

   The actual current RMS can be calculated using the ratio factor and the value of the register RMSIA. By default, the current RMS should be with two decimal places.

2. Use the same method to calculate the ratio factors of the RMS values of voltage channels.

All above parameters are preset when designing the energy meters. They should be constant

when calibrating the meters.

## 8.17.6.7. Calibrating an Energy Meter

**1. Power Gain Calibration of Phase A**

1.1 Conditions

To select Phase A for energy metering with power factor 1.0 when powering on the calibration equipment with 100%$I_n$ and 100%$U_n$.

1.2 Calculation

Equation 8-7

1.3 Related registers

See Table 8-19 for details.

The active power gain calibration register of Phase A is SCPA, and the registers in segments are SCPA0 (0x2A9E), SCPA1 (0x2A9F), SCPA2 (0x2AA0).

When power gain calibration in segments is disabled, the value of the register SCPA0 is written into SCPA; when power gain calibration in segments is enabled, the calibrated current RMS (IRMS) is used to determine the segment to be used. For example, when IRMS<SCPTH0A, the value of the register SCPA0 is written into SCPA; when SCPTH0A<IRMS<SCPTH1A, the value of the register SCPA1 is written into SCPA; when IRMS>SCPTH1A, the value of the register SCPA2 is written into SCPA.

If the error of this energy meter is 4.5%, and the value of the register SCPA is 0x11345682,

$$S = 2^{31}(\frac{1}{1+e} - 1) + S_1(\frac{1}{1+e})$$
$$= 2^{31} \times (\frac{1}{1+0.045} - 1) + 0x11345682 \times (\frac{1}{1+0.045})$$
$$= 0xFFFA7CF014 + 0x1076AD6C = 0xAF39D80$$

**Note:** Use the registers PECEN and AECEN (MTPARA2, 0x2002) to disable the power gain calibration and phase compensation in segments.

**2. Phase Compensation of Phase A**

2.1 Conditions

To select Phase A for energy metering with power factor 0.5L when powering on the calibration equipment with 100%$I_n$ and 100%$U_n$.

2.2 Calculation

Equation 8-8

2.3 Related registers

See Table 8-13 for details.

The phase compensation value of Phase A is in bit7~bit0 of AECx (x=0~4), of which, bit7 is the sign bit, which should be set to 1, meaning the compensation value is negative.

When phase compensation in segments is disabled, the phase compensation value (with the sign) is written into bit7~bit0 of the register AEC0; when phase compensation in segments is enabled, the calibrated current RMS (IRMS) is used to determine which segment should be used. For example, when AEC0A<IRMS<AEC1A, the phase compensation value (with the sign) is written into bit7~bit0 of the register AEC1.

When the phase error is 0.36%, and $f_{ADC}$=819200Hz,

$$N = \frac{3011}{2} \times Error \times \frac{f_{ADC}}{819200} = \frac{3011}{2} \times 0.0036 \times \frac{819200}{819200} = 0x5$$

Bit7 must be set to 1 to lag the current signal, so 0x85 is written into the register finally.

Note: Clear the register AEC0 firstly, and then, compensate phase.

**3. Calibrating gains and compensating phase in Phase B and Phase C.**

**4. Detecting the energy error on the overall system. If the error is not the right one, repeat the above steps.**

**5. Calibrating current RMS of Phase A**

5.1 Conditions

To power on the calibration equipment with 100%$I_n$ and 100%$U_n$.

5.2 Calculation

Equation 8-10 and Equation 8-10

5.3 Related register

SCIA (0x2AAD)

When $I_n$, is 1.501A, $I_1$, is 1.343A, and the value of the register SCIA is 0x21124562,

$$e = \frac{I_1 - I_n}{I_n} = \frac{1343 - 1501}{1501} = -10.5\%$$

and then,

$$S = 2^{31}(\frac{1}{1+e} - 1) + S_1(\frac{1}{1+e})$$
$$= 2^{31} \times (\frac{1}{1 - 0.105} - 1) + 0x21124562 \times (\frac{1}{1 - 0.105})$$
$$= 0xF044A5E + 0x24F3854C$$
$$= 0x33F7CFAA$$

**6. Calibrating voltage RMS of Phase A**

6.1 Conditions

To power on the calibration equipment with 100%$I_n$ and 100% $U_n$.

6.2 Calculation

Equation 8-11 and Equation 8-7

6.3 Related register

SCUA (0x2AAC)

When the voltage RMS, $U_n$, is 220.1V, the RMS value $U_1$ is 230.04V, and the value of the register SCUA is 0x10035251,

$$e = \frac{U_1 - U_n}{U_n} = \frac{23004 - 22010}{22010} = 4.5\%$$

and then,

$$S = 2^{31}(\frac{1}{1+e} - 1) + S_1(\frac{1}{1+e})$$
$$= 2^{31} \times (\frac{1}{1 + 0.045} - 1) + 0x10035251 \times (\frac{1}{1 + 0.045})$$
$$= 0xFFFA7CF014 + 0xF52CBB5$$
$$= 0x9CFBBC9$$

Use the same method to calibrate the energy gain, phase and RMS value of Phase B and Phase C.

# 8.17.7.Working Status Control of the Metering Block

The energy metering DSP unit is composed of sub-processing-units, including Hilbert processing unit and three energy metering processing units, Energy Metering Processing Unit A (MPA), Energy Metering Processing Unit B (MPB) and Energy Metering Processing Unit C (MPC). Every register has an SRAM, distributing in the space located in 0x2000~0x27FF. After reset, the data in the SRAM are in random, which should be cleared and configured.

The registers, MPA Clear Control Register (CLRMPA), MPB Clear Control Register (CLRMPB), MPC Clear Control Register (CLRMPC) and Hilbert Clear Control Register

(CLRHLB) are used to clear all input data in the SRAM. All SRAM addresses can be accessed in about 20ms, and all data in it is cleared.

The energy metering processing units share the SRAM with MCU, but the former one has priority. To improve the access rate of MCU to SRAM, and the initialization of the energy metering DSP, the energy metering processing units should be turned off firstly via configuring the registers, MPA Sleep Control Register (SLPMPA), MPB Sleep Control Register (SLPMPB), MPC Sleep Control Register (SLPMPC) and Hilbert Sleep Control Register (SLPHLB) to permit MCU use SRAM independently. But when metering energy, this operation is not allowed.

Keep the energy tanks clear via configuring Energy Tank Clear Control Register (CLRACC), to prevent them from metering unnecessary energy data which are generated before the metering module works normally.

### 8.17.8.Meter Type Configuraton

V9003/V9103 is designed via configuring Energy Metering Mode Control Register (MTMODE) into 3-phase meters for both 3-wire and 4-wire service.

## 8.18. Turning on the Metering Module

After reset, the clock (Clock 2) for energy metering DSP is turned off, and all energy metering configuration registers are set to 0. It can be turned on to work as follows:

1. To configure and turn on PLL, ADC and related analog modules;
2. To configure MCU to use PLL clock as the working frequency;
3. To configure the clock frequency for the energy metering DSP to 6.5536MHz;
4. To write 0x20000000 into the buffer registers located in 0x3004~0x3007;
5. To turn on Clock 2;
6. To write the data in the buffer registers (0x3004~0x3007) into the register MTPARA0 (0x2000) in 48 clock cycles;
7. To wait 20ms;
8. To set the registers CLRMPA, CLRMPB, CLRMPC and CLRHLB to 1, and clear all internal memories of the energy metering DSP in 20ms;
9. To set the registers SLPMPA, SLPMPB, SLPMPC and SLPHLB to 1, and set CLRMPA, CLRMPB and CLRHLB to 0;

10. To configure the parameters of gain calibration, phase compensation and threshold; to configure the return difference, PGA, meter type, CF source selection, and no-load detection. Set CLRACC to 1;
11. To configure the register ADCON. Turn on the corresponding ADCs based on the meter type, and wait 1s to ensure the convergence of the filter;
12. To configure energy metering configuration registers. Set CLRACC to 0, set CLRMPC to 1, to enable energy accumulation, and configure the register CFON to turn on CF pulse output;
13. The energy metering DSP starts to output CF pulses and CF interrupts.

# 8.19. Analog Control Registers

Table 8-33 lists all analog control registers. By default, all bits are set to 0.

**Table 8-33 Analog Control Registers**

| Addr | Register | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
|---|---|---|---|---|---|---|---|---|---|
| 0x1B00 | AGIA | | | | | | GIA<2> | GIA<1> | GIA<0> |
| 0x1B01 | AGIB | | | | | | GIB<2> | GIB<1> | GIB<0> |
| 0x1B02 | AGIC | | | | | | GIC<2> | GIC<1> | GIC<0> |
| 0x1B03 | AGIN | | | | | | GIN<2> | GIN<1> | GIN<0> |
| 0x1B04 | AGUA | | | | | | | | GUA |
| 0x1B05 | AGUB | | | | | | | | GUB |
| 0x1B06 | AGUC | | | | | | | | GUC |
| 0x1B07 | AGM | | | REFGM <1> | REFGM <0> | | | | GM |
| 0x1B08 | SHORT | SIGIN ENU | SIGIN ENI | TRB <1> | TRB <0> | SHORT V2 | SHORT V1 | SHORT I2 | SHORT I1 |
| 0x1B09 | SIGCLK | | | | | | | F1K | IDCEN |
| 0x1B0A | ADRST | ADRSTM | ADRS TUC | ADRS TUB | ADRS TUA | ADRS TIN | ADRSTIC | ADRSTIB | ADRSTIA |
| 0x1B0B | ICtrl0 | | REFGITI | REFBITI | | ADIT2I <1> | ADIT2I <0> | ADIT1I <1> | ADIT1I <0> |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0x1B0C | UCtrl | | REFGITU | REFBITU | | ADIT2U<1> | ADIT2U<0> | ADIT1U<1> | ADIT1U<0> |
| 0x1B0D | ICtrl1 | RESDIV | REFGITM | REFBITM | | ADIT2M<1> | ADIT2M<0> | ADIT1M<1> | ADIT1M<0> |
| 0x1B0E | BGPCFG | | REST<2> | REST<1> | REST<0> | | | BGPCHOPN | |
| 0x1B0F | CURRCFG | | | | | CURRLP | RTRIM<2> | RTRIM<1> | RTRIM<0> |
| 0x1B10 | CtrlBAT | | | | | BATDISC<1> | BATDISC<0> | | |
| 0x1B11 | ADCCtrl | ADPDMN | ADPDUCN | ADPDUBN | ADPDUAN | ADPDINN | ADPDICN | ADPDIBN | ADPDIAN |
| 0x1B12 | PLLCtrl | | | PLLPDN | BGPPDN | | | | |
| 0x1B13 | CLKCFG | | | CLKOSEL | ADCLKEXT | XEASYN | XTRIM<2> | XTRIM<1> | XTRIM<0> |
| 0x1B14 | LDOCFG | V3P | | LDO33PD | V3P5 | | | LDOVSEL<1> | LDOVSEL<0> |
| 0x1B15 | FRQCFG0 | MEAS<2> | MEAS<1> | MAES<0> | PLLSEL | ADCLKSEL<1> | ADCLKSEL<0> | MCUCLKSEL<1> | MCUCLKSEL<0> |
| 0x1B16 | FRQCFG1 | | MEACLKSEL<1> | MEACLKSEL<0> | WDCKSEL | | | PLLIB<1> | PLLIB<0> |
| 0x1B17 | IRCtrl | | | | | IRPDN | IRSET | IRIT<1> | IRIT<0> |

The bit definition of some registers is listed as follows.

**Table 8-34 Bit Definition of SHORT**

| Bit | | Description | Remark |
|---|---|---|---|
| Bit0 | SHORTI1 | To enable or disable the short circuit before the current channel ADCs | 1, to enable; By default, 0. |

| | | | |
|---|---|---|---|
| Bit1 | SHORTI2 | To enable or disable the short circuit before the current amplifies | 1, to enable; By default, 0. |
| Bit2 | SHORTV1 | To enable or disable the short circuit before the voltage channel ADCs | 1, to enable; By default, 0. |
| Bit3 | SHORTV2 | To enable or disable the short circuit before the voltage amplifiers | 1, to enable; By default, 0. |
| Bit[5:4] | TRB<1:0> | To enable or disable the internal biasing in ADCs | 11, to enable; By default, 00. |
| Bit6 | SIGINENI | To enable or disable internal signal source of current channel ADCs | 1, to enable; By default, 0. |
| Bit7 | SIGINENU | To enable or disable internal signal source of voltage channel ADCs | 1, to enable; By default, 0. |

**Table 8-35 Bit Description of SIGCLK**

| Bit | | Description | Remark |
|---|---|---|---|
| Bit0 | IDCEN | To enable or disable the increase of input common mode voltage of ADCs | 1, to enable; By default, 0. |
| Bit1 | F1K | To enable or disable 1kHz as the internal signal source | 1, to enable; By default, 0. |

**Table 8-36 Bit Description of ADRST**

| Bit | | Description | Remark |
|---|---|---|---|
| Bit0 | ADRSTIA | To enable or disable the reset integrator of Channel IA modulator | 1, to enable; By default, 0. |
| Bit1 | ADRSTIB | To enable or disable the reset integrator of Channel IB modulator | 1, to enable; By default, 0. |
| Bit2 | ADRSTIC | To enable or disable the reset integrator of Channel IC modulator | 1, to enable; By default, 0. |
| Bit3 | ADRSTIN | To enable or disable the reset integrator of Channel IN modulator | 1, to enable; By default, 0. |
| Bit4 | ADRSTUA | To enable or disable the reset integrator of Channel UA modulator | 1, to enable; By default, 0. |
| Bit5 | ADRSTUB | To enable or disable the reset integrator of Channel UB modulator | 1, to enable; By default, 0. |

| | | | |
|---|---|---|---|
| Bit6 | ADRSTUC | To enable or disable the reset integrator of Channel UC modulator | 1, to enable; By default, 0. |
| Bit7 | ADRSTM | To enable or disable the reset integrator of Channel M modulator | 1, to enable; By default, 0. |

**Table 8-37 Bit Description of ICtrl0**

| Bit | | Description | Remark |
|---|---|---|---|
| Bit[1:0] | ADIT1I<1:0> | To adjust the biasing current by the 1st modulator of current channels | 00, ×1; 01, ×1.5; 10, ×2; 11, ×2.5<br>By default, 00. |
| Bit[3:2] | ADIT2I<1:0> | To adjust the biasing current by the 2nd modulator of current channels | 00, ×1; 01, ×1.5; 10, ×2; 11, ×2.5<br>By default, 00. |
| Bit5 | REFBITI | To adjust the reference buffer biasing current by the current channel ADC | 0, ×1; 1, ×1.33<br>By default, 00. |
| Bit6 | REFGITI | To adjust the reference generation biasing current by the current channel ADC | 0, ×1; 1, ×1.5<br>By default, 0. |

**Table 8-38 Bit Description of UCtrl**

| Bit | | Description | Remark |
|---|---|---|---|
| Bit[1:0] | ADIT1U<1:0> | To adjust the biasing current by the 1st modulator of voltage channels | 00, ×1; 01, ×1.5; 10, ×2; 11, ×2.5<br>By default, 00. |
| Bit[3:2] | ADIT2U<1:0> | To adjust the biasing current by the 2nd modulator of voltage channels | 00, ×1; 01, ×1.5; 10, ×2; 11, ×2.5<br>By default, 00. |
| Bit5 | REFBITU | To adjust the reference buffer biasing current by the voltage channel ADC | 0, ×1; 1, ×1.33<br>By default, 0. |
| Bit6 | REFGITU | To adjust the reference generation biasing current by the voltage channel ADC | 0, ×1; 1, ×1.5<br>By default, 0. |

**Table 8-39 Bit Description of IRCtrl**

| Bit | | Description | Remark |
|---|---|---|---|
| Bit[1:0] | IRIT<1:0> | To adjust biasing current in the IR module | 00, ×0.5; 01, ×0.67; 10, ×0.75; 11, ×1 |
| Bit2 | IRSET | To enable or disable bypassing IR receivers | 0, to enable |
| Bit3 | IRPDN | To disable IR receivers | 0, to disable. By default, 0. |

# 9. Interrupt

POR/BOR, RST Pin Reset, IO/RTC Wake-up Reset, WDT Reset, and Debug Reset all can reset the registers of the interrupt control circuit into default state. In Sleep or Deep Sleep, the interrupt control circuit stops working and gets into low power consumption mode.

In V9003/V9103, the system provides 9 timer interrupts (including: Timer0/1/2 overflow interrupt, TimerA/B overflow interrupts, TimerA/B compare/capture interrupts), 2 IO interrupts (IO Interrupt 0/1), receiving/transmitting interrupt of UART0~UART5, 5 P3 port interrupt, RTC second interrupt and RTC illegal interrupt, Flash page erase/programming and mass erase interrupt, active/reactive CF pulse interrupt, power-up interrupt, power-down interrupt, IR interrupt, and PLL lock interrupt. When an IO port is used for IO Interrupt 0/1, users should configure the GPIO for special function.

Figure 9-1 and Figure 9-2 show the architecture of the interrupt control circuit, including the interrupt enabling, priority control, and interrupt flags.

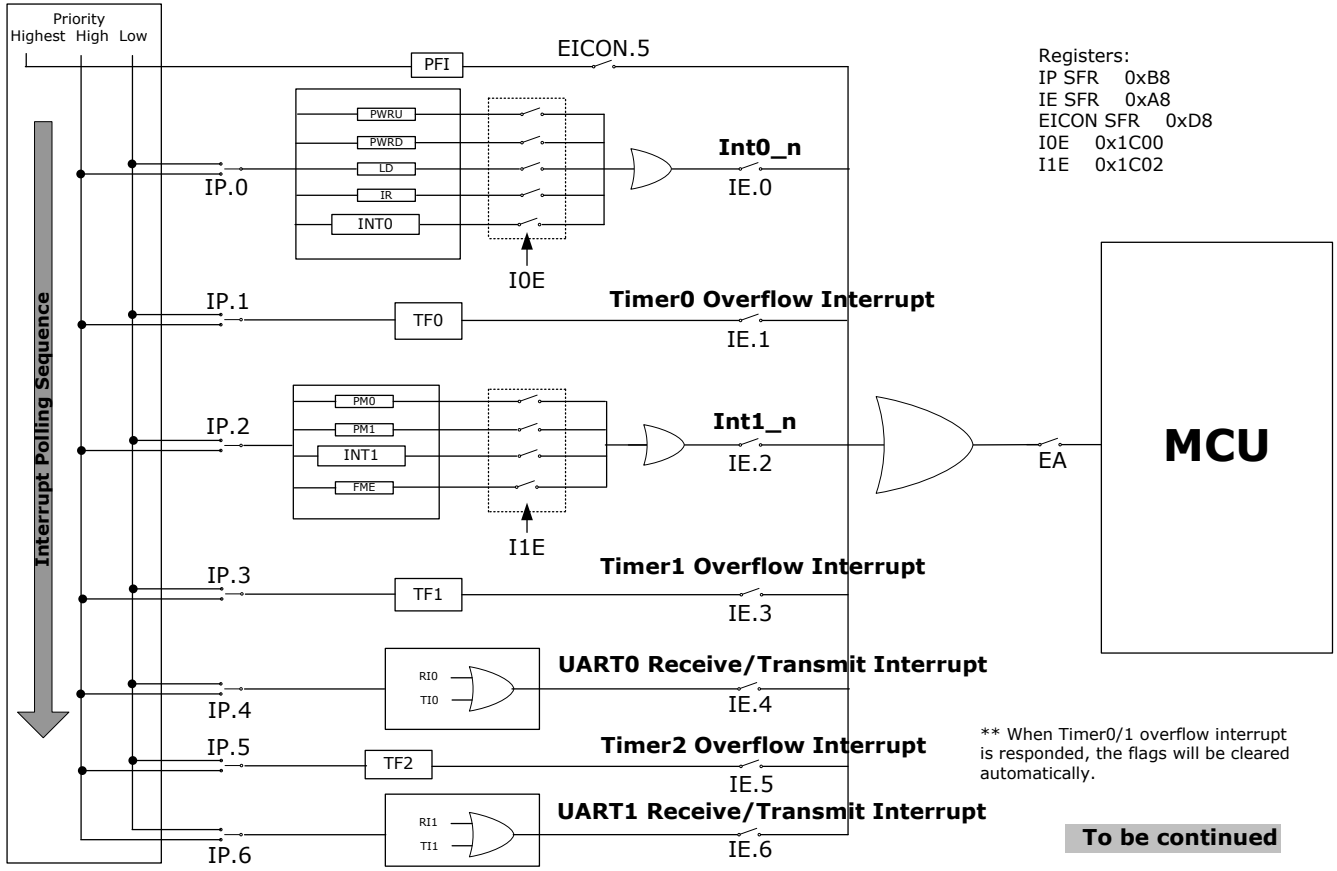



**Figure 9-1 Interrupt Architecture I**

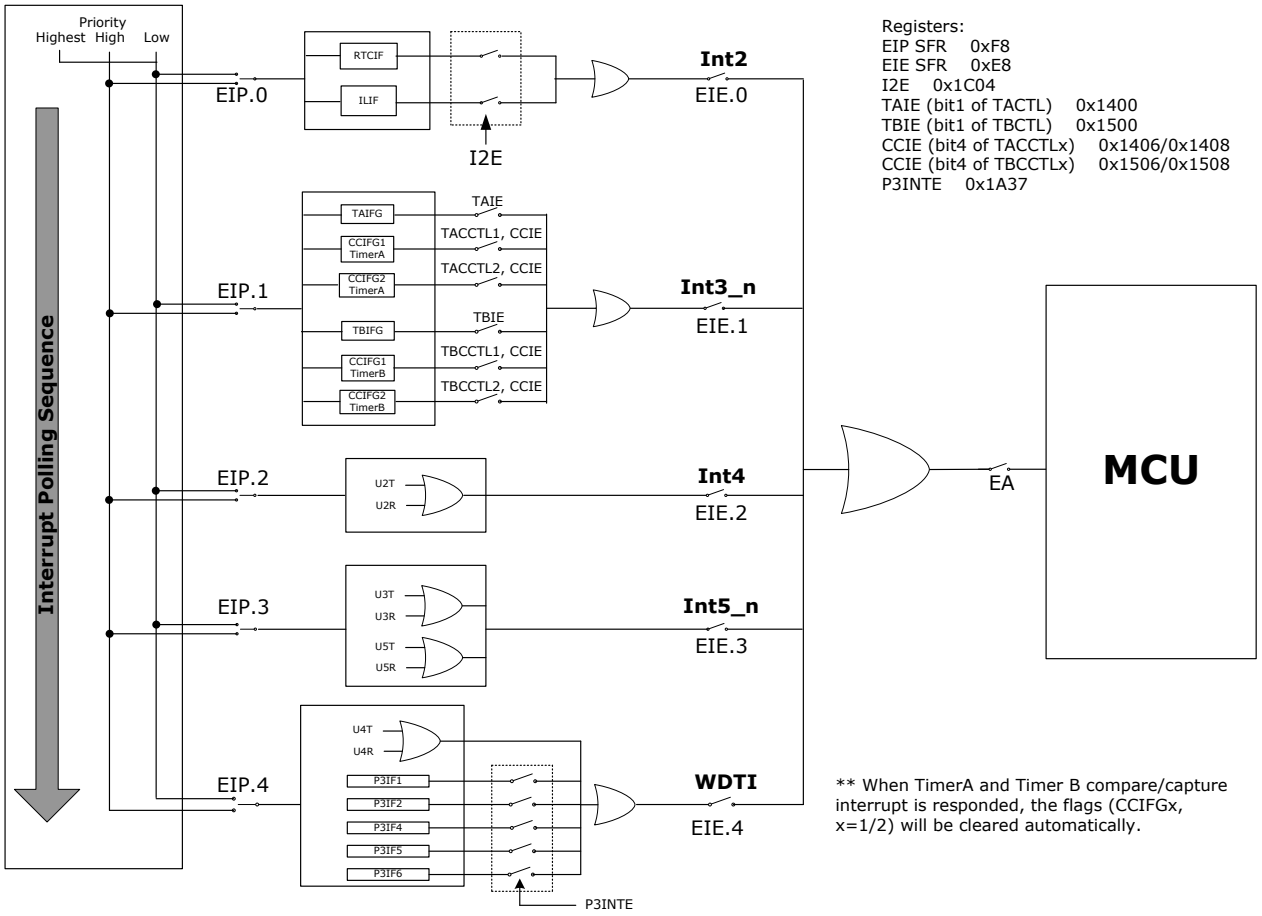



**Figure 9-2 Interrupt Architecture II**

# 9.1. Special Function Registers (SFR)

In V9003/V9103, the special function registers in the interrupt control circuit includes: IE SFR (0xA8), IP SFR (0xB8), EXIF SFR (0x91), EICON SFR (0xD8), EIE SFR (0xE8), and EIP SFR (0xF8). These registers have the capability of interrupt enabling, priority control and interrupt flags.

## 9.1.1. IE SFR (0xA8)

**Table 9-1 IE SFR (0xA8)**

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Description | EA | ES1 | ET2 | ES0 | ET1 | EX1 | ET0 | EX0 |
| Bit Address | AFh | AEh | ADh | ACh | ABh | AAh | A9h | A8h |

**Table 9-2 Bit Description of IE SFR**

| Bit | Description |
|---|---|
| IE.7 | EA – Global interrupt enable bit. EA=0, to disable all interrupts (EA overrides individual interrupt enable bits). EA=1, each interrupt is enabled or masked by its individual enable bit. |
| IE.6 | ES1 –UART1 interrupt enable bit. ES1=0, to disable UART 1 Interrupt (TI1 and RI1). ES1=1, to enable the interrupt generated by the flag TI1 or RI1. |
| IE.5 | ET2 –Timer 2 interrupt enable bit. ET2=0, to disable Timer 2 interrupt (TF2). ET2=1, to enable the interrupt generated by the flag TF2. |
| IE.4 | ES0- UART0 interrupt enable bit. ES0=0, to disable UART0 interrupt (TI0 and RI0). ES0=1, to enable the interrupt generated by the flag TI0 or RI0. |
| IE.3 | ET1 –Timer 1 interrupt enable bit. ET1=0, to disable Timer 1 interrupt (TF1). ET1=1, to enable the interrupt generated by the flag TF1. |
| IE.2 | EX1 – Int1_n interrupt enable bit. EX1 = 0, to disable Int1_n interrupt. EX1 = 1, to enable Int1_n interrupt. |
| IE.1 | ET0 –Timer 0 interrupt enable bit. ET0 = 0, to disable Timer 0 interrupt (TF0). ET0 = 1, to enable the interrupt generated by the flag TF0. |
| IE.0 | EX0 –Int0_n interrupt enable bit. EX0 = 0, to disable Int0_n interrupt. EX0 = 1, to enable Int0_n interrupt. |

The bit EA in the IE SFR (IE.7) is a global enable bit for all interrupts. When EA = 1, each interrupt

is enabled/disabled by its individual enable bit. When EA = 0, all interrupts are disabled.

## 9.1.2. EIE SFR (0xE8)

**Table 9-3 Bit Description of EIE SFR (0xE8)**

| Bit | Description |
|---|---|
| EIE.7~5 | Reserved, read out as 1. |
| EIE.4 | EWDI – WDTI interrupt enable bit. 0, to disable; 1, to enable. |
| EIE.3 | EX5 – Int5_n interrupt enable bit. 0, to disable; 1, to enable. |
| EIE.2 | EX4 – Int4 interrupt enable bit. 0, to disable; 1, to enable. |
| EIE.1 | EX3 –Int3_n interrupt enable bit. 0, to disable; 1, to enable. |
| EIE.0 | EX2 – Int2 interrupt enable bit. 0, to disable; 1, to enable. |

## 9.1.3. EXIF SFR (0x91)

**Table 9-4 Description of EXIF SFR (0x91)**

| Bit | Description |
|---|---|
| EXIF.7 | IE5 – Int5_n interrupt flag. IE5=1, to indicate that interrupt Int5_n was triggered. Setting IE5 by program generates an interrupt, if the interrupt is enabled. |
| EXIF.6 | IE4 – Int4 interrupt flag. IE4=1, to indicate that interrupt Int4 was triggered. Setting IE4 by program generates an interrupt, if the interrupt is enabled. |
| EXIF.5 | IE3 – Int3_n interrupt flag. IE3=1, to indicate that interrupt Int3_n was triggered. Setting IE3 by program generates an interrupt, if the interrupt is enabled. |
| EXIF.4 | IE2 – Int2 interrupt flag. IE2=1, to indicate that interrupt Int2 was triggered. Setting IE2 by program generates an interrupt, if the interrupt is enabled. |
| EXIF.3 | Reserved, read out as 1. |
| EXIF.2 ~ 0 | Reserved, read out as 0. |

## 9.1.4. EICON SFR (0xD8)

**Table 9-5 Bit Description of EICON SFR (0xD8)**

| Bit | Description |
|---|---|
| EICON.7 | SMOD1 – UART1 baud rate double enable bit. When SMOD1=1, the baud rate for UART1 is doubled. |
| EICON.6 | Reserved, read out as 1. |
| EICON.5 | EPFI – PFI enable bit. EPFI=0, to disable; EPFI=1, to enable. |
| EICON.4 | PFI – PFI interrupt flag. The source of PFI is the page-erase and page-programming signals from the Flash interface. Both signals are combined to be a 1-period wide synchronous pulse, which can be extended into a 4-period wide pulse. This interrupt occurs in IAP. When erasing or writing Flash, the processor must standby to ensure the accuracy when reading Flash. When erasing or writing Flash is finished, a PFI is triggered to wake up the processor, and then, further operation is executed. |
| EICON.3 | WDTI – WDTI interrupt flag. WDTI =1, to indicate that a WDTI interrupt was triggered. This flag must be cleared by software before exiting from the interrupt service routine. Otherwise, the interrupt occurs again. Setting this flag by program triggers a WDTI interrupt, if the interrupt is enabled. |
| EICON.2~0 | Reserved, read out as 0. |

In V9003/V9103, there are two tiers of interrupts: Interrupt Priority 1 and Interrupt Priority 0. Both IP SFR and EIP SFR can configure the priority of interrupts. Interrupt Priority 1 takes precedence over Interrupt Priority 0. All interrupts can be assigned either interrupt priority.

In addition to an assigned priority level (Interrupt Priority 1 or 0), each interrupt has a polling sequence (see "Interrupt Resources", a smaller polling sequence number means an interrupt with higher polling priority). If two interrupt of different priority occurs, the one of Interrupt Priority 1 is serviced first. If two interrupts of the same tier (Interrupt Priority 1 or Interrupt Priority 0) occur simultaneously, the polling sequence of them is observed. Once an interrupt is being serviced, only an interrupt of higher priority level can interrupt the service routine of the interrupt currently being serviced (see "Interrupt Processing" for details).

Registers IP SFR and EIP SFR control the priority of all interrupts.

## 9.1.5. IP SFR (0xB8)

**Table 9-6 IP SFR (0xB8)**

| BIT | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Description | Reserved | PS1 | PT2 | PS0 | PT1 | PX1 | PT0 | PX0 |

| Bit Address | BFh | BEh | BDh | BCh | BBh | BAh | B9h | B8h |
|---|---|---|---|---|---|---|---|---|

**Table 9-7 Bit Description of IP SFR**

| Bit | Description |
|---|---|
| IP.7 | Reserved. Read out as 1. |
| IP.6 | PS1 – UART 1 interrupt priority control bit. PS1=0, to set UART 1 interrupt (TI1 or RI1) to Interrupt Priority 0. PS1=1, to set UART 1 interrupt (TI1 or RI1) to Interrupt Priority 1. |
| IP.5 | PT2 – Timer 2 interrupt priority control bit. PT2=0, to set Timer 2 interrupt (TF2) to Interrupt Priority 0. PT2=1, to set Timer 2 interrupt (TF2) to Interrupt Priority 1. |
| IP.4 | PS0 – UART 0 interrupt priority select bit. PS0=0, to set UART 0 interrupt (TI0 or RI0) to Interrupt Priority 0. PS0=1, to set UART 0 interrupt (TI0 or RI0) to Interrupt Priority 1. |
| IP.3 | PT1 – Timer 1 interrupt priority control bit. PT1=0, to set Timer 1 interrupt (TF1) to Interrupt Priority 0. PT1=1, to set Timer 1 interrupt (TF1) to Interrupt Priority 1. |
| IP.2 | PX1 – Int1_n interrupt priority control bit. PX1=0, to set Int1_n interrupt to Interrupt Priority 0. PX1=1, to set Int1_n interrupt to Interrupt Priority 1. |
| IP.1 | PT0 – Timer 0 interrupt priority control bit. PT0=0, to set Timer 0 interrupt (TF0) to Interrupt Priority 0. PT0=1, to set Timer 0 interrupt (TF0) to Interrupt Priority 1. |
| IP.0 | PX0 – Int0_n interrupt priority control bit. PX0=0, to set Int0_n interrupt to Interrupt Priority 0. PX0=1, to set Int0_n interrupt to Interrupt Priority 1. |

## 9.1.6. EIP SFR (0xF8)

**Table 9-8 Bit Description of EIP SFR**

| Bit | Description |
|---|---|
| EIP.7~5 | Reserved, read out as 1. |
| EIP.4 | PWDI – WDT interrupt priority select bit. 0, to set the interrupt to Interrupt Priority 0; 1, to set the interrupt to Interrupt Priority 1. |
| EIP.3 | PX5 – Int5_n interrupt priority select bit. 0, to set the interrupt to Interrupt Priority 0. 1, to set the interrupt to Interrupt Priority 1, |
| EIP.2 | PX4 – Int4 interrupt priority select bit. 0, to set the interrupt to Interrupt Priority 0. 1, to set the interrupt to Interrupt Priority 1. |
| EIP.1 | PX3 – Int3_n interrupt priority select bit. 0, to set the interrupt to Interrupt Priority 0. 1, to set the interrupt to Interrupt Priority 1. |

| | |
|---|---|
| EIP.0 | PX2 – Int2 interrupt priority select bit. 0, to set the interrupt to Interrupt Priority 0. 1, to set the interrupt to Interrupt Priority 1. |

# 9.2. Interrupt Processing

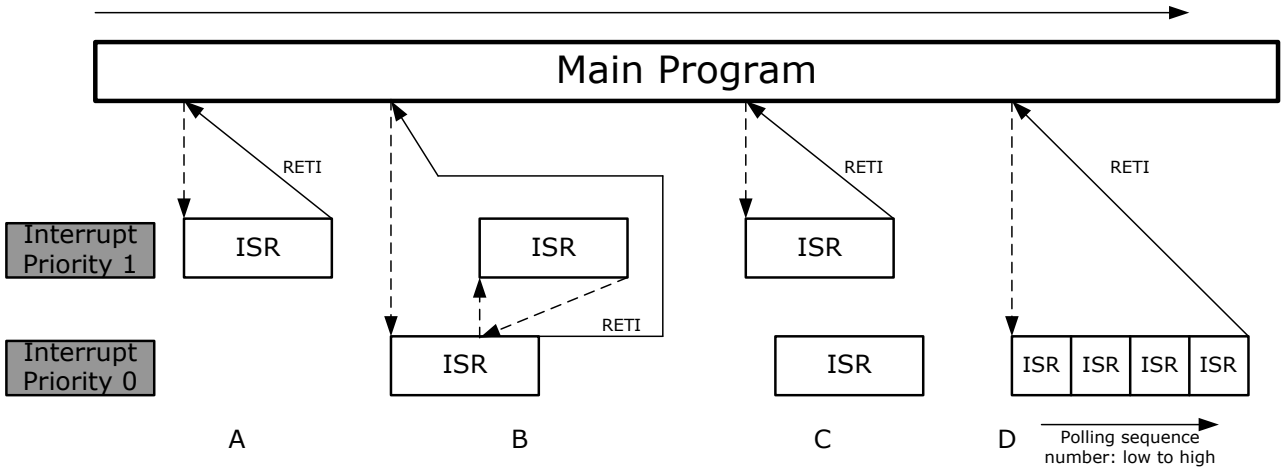



**Figure 9-3 Processing Interrupts**

Figure 9-3 shows the interrupt processing in V9003/V9103. When an enabled interrupt occurs,

- The CPU jumps to the interrupt vector address to execute the interrupt service routine (ISR) associated with that interrupt. The CPU completes the ISR execution unless another interrupt of higher priority occurs. Each ISR ends with an RETI (return from interrupt) instruction, as shown in A of Figure 9-3 Processing Interrupts.

- After executing the RETI, the CPU returns to the next instruction that would have been executed if the interrupt had not occurred. The CPU always completes the instruction in progress before servicing an interrupt. If the instruction in progress is RETI, or a write access to any of the registers, IP SFR, IE SFR, EIP SFR, or EIE SFR, the CPU completes one additional instruction before servicing the ISR.

- In V9003/V9103, Interrupt Priority 1 has higher priority than Interrupt Priority 0. So, the ISR of Interrupt Priority 0 only can be interrupted by the ISR of Interrupt Priority 1, as shown in B and C of Figure 9-3.

- An ISR of Interrupt Priority 0 was intruded by one of Interrupt Priority 1. When the latter one is executed, the program will return to execute the unfinished ISR of Interrupt Priority 0, and then, execute the instruction RETI to finish the ISR, as shown in B of Figure 9-3.

- When two interrupts of the same tier (Interrupt Priority 1 or Interrupt Priority 0) occur simultaneously, the polling sequence of them is observed, as shown in D of Figure 9-3.

- Interrupt latency depends on the current state of the CPU.
  - The shortest interrupt latency is five instruction cycles: one to detect the interrupt, and four to perform the LCALL to the ISR.
  - The longest latency (thirteen instruction cycles) occurs when the CPU is currently executing an RETI instruction followed by a MUL or DIV instruction. The thirteen instruction cycles in this case are: one to detect the interrupt, three to complete the RETI, five to execute the DIV or MUL, and four to execute the LCALL to the ISR. For the maximum latency case, the interrupt latency is 13x4 = 52 clock cycles.
- To ensure that edge-sensitive interrupts are detected, the level on the corresponding ports should be held high for four clock cycles and then low for four clock cycles. Level-sensitive interrupts are not latched, so they must remain active until being serviced.

# 9.3. Extended Interrupts

The extended interrupt module has the following functions:

1. Several input interrupt signals are output as one interrupt signal, which can be realized easily via logic "or";
2. When a group of various input interrupts are output as one interrupt, the CPU must be ensured to distinguish every one of it, and no one gets lost even if they are processed simultaneously or two interrupts are processed in a short time. Every input interrupt has a dedicated flag register, which can be accessed by CPU to detect whether an interrupt occurs or not. If the flag is set, the extended interrupt module will generate interrupt repeatedly to ask for being processed by CPU;
3. In 8051 MCU, the width of an interrupt signal must be more than 4 periods. So all input interrupt signals should be output as 4-period wide interrupts.

### 9.3.1. PFI

The source of PFI is the page erase and page programming signal of FLASH module. These two signals are integrated into one synchronous 1-period wide pulse. The extended interrupt module extends this pulse into 4 periods.

PFI is mainly performed in IAP. When erasing or writing of FLASH, CPU should be standby to protect reading FLASH from some errors.

PFI interrupt has the highest interrupt priority. If EA=1 (bit7 of IE SFR) and EPFI=1 (bit5 of EICON SFR), when the flag PFI (bit4 of EICON SFR) is set, a PFI interrupt is generated to MCU.

### 9.3.2. Int0_n

The interrupt Int0_n is extended into 5 sub-interrupts as shown in Table 9-9. Every sub-interrupt has its enable bit and interrupt flag, among which, the only-read register I0F can sign all interrupt flags, and the registers I0SF0~I0SF4, which can be read and written, can set every sub-interrupt flag respectively. When interrupt flags are set to 1, and the corresponding interrupts are enabled, a 4-period wide pulse is sent periodically by the extended interrupt module on Int0_n interrupt.

**Table 9-9 Int0_n**

| No. | Interrupt Source | Description | Enable Bit | Interrupt Flag (Read) | Interrupt Flag (R/W) |
|---|---|---|---|---|---|
| 0 | PRUP | A power-up signal, to indicate that the input voltage is higher than the threshold value, for example, 2V. It was synchronized, and is triggered on the rising edge. A reset signal is generated when powering up. So, it is not needed actually. | I0E.0 | I0F.0 | I0SF0.0 |
| 1 | PRDOWN | A power-down signal, to indicate that the input voltage is lower than the threshold value, 1.8V. It was synchronized, and to ensure the interrupt can be received, it is set to a level-triggered interrupt. | I0E.1 | I0F.1 | I0SF1.0 |
| 2 | PLLLOCK | A PLL-locked signal on high level, to indicate that PLL is locked. It was synchronized. It is generated on | I0E.2 | I0F.2 | I0SF2.0 |

| | | | | | |
|---|---|---|---|---|---|
| | | rising edge or falling edge. | | | |
| 3 | IR_RI | An IR input signal, an asynchronous signal, being triggered on the falling edge. | I0E.3 | I0F.3 | I0SF3.0 |
| 4 | EX_INT0 | IO Interrupt 0, asynchronous, low active, triggered on the falling edge. | I0E.4 | I0F.4 | I0SF4.0 |

Table 9-10 to Table 9-16 list all related registers to this interrupt.

**Table 9-10 Int0_n Interrupt Enable Register (I0E, 0x1C00)**

| 0x1C00, R/W | Int0_n Interrupt Enable Register, I0E | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | EXI0N | IR | LD | PWRD | PWRU |
| Default Value | - | - | - | 0 | 0 | 0 | 0 | 0 |

- When EA=1 (IE.7), and EX0=1 (IE.0), all sub-interrupts of Int0_n can be enabled or disabled via configuring the register I0E.
- 1, to enable the interrupt; 0, to disable the interrupt.

**Table 9-11 Int0_n Interrupt Flag Register (I0F, 0x1C01)**

| 0x1C01, R | Int0_n Interrupt Flag Register, I0F | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | EXI0N | IR | LD | PWRD | PWRU |
| Default Value | - | - | - | 0 | 0 | 0 | 0 | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

**Table 9-12 Powerup Interrupt Flag Register (I0SF0, 0x1C10)**

| 0x1C10, R/W | PRUP Interrupt Flag Register, INT0_SCIF0 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | PWRU |
| Default Value | - | - | - | - | - | - | - | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

**Table 9-13 Powerdown Interrupt Flag Register (I0SF1, 0x1C11)**

| 0x1C11, R/W | PWDOWN Interrupt Flag Register, I0SF1 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | PWRD |
| Default Value | - | - | - | - | - | - | - | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

**Table 9-14 PLL Lock Interrupt Flag Register (I0SF2, 0x1C12)**

| 0x1C12, R/W | PLL LOCK Interrupt Flag Register, I0SF2 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | LD |
| Default Value | - | - | - | - | - | - | - | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

**Table 9-15 IR Interrupt Flag Register (I0SF3, 0x1C13)**

| 0x1C13, R/W | IR Interrupt Flag Register, I0SF3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | IR |
| Default Value | - | - | - | - | - | - | - | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

**Table 9-16 IO Interrupt 0 Flag Register (I0SF4, 0x1C14)**

| 0x1C14, R/W | IO Interrupt 0 Flag Register, I0SF4 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | EXI0N |
| Default Value | - | - | - | - | - | - | - | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

## 9.3.3. Int1_n

The interrupt Int1_n is extended into 4 sub-interrupts, as shown in Table 9-17. Every sub-interrupt has its enable bit and interrupt flag, among which, the only-read register I1F can

sign all interrupt flags, and the registers I1SF0/1/4/5, which can be read and written, can set all sub-interrupt flags respectively. When the interrupt flags are set to 1, and the corresponding interrupts are enabled, a 4-period wide pulse is sent periodically by the extended interrupt module on Int1_n interrupt.

**Table 9-17 Int1_n**

| No. | Interrupt Source | Description | Enable Bit | Interrupt Flag (Read) | Interrupt Flag (R/W) |
|---|---|---|---|---|---|
| 0 | PM_0 | Active CF interrupt, a synchronous 1-period wide pulse. | I1E.0 | I1F.0 | I1SF0.0 |
| 1 | PM_1 | Reactive CF interrupt, a synchronous 1-period wide pulse. | I1E.1 | I1F.1 | I1SF1.0 |
| 2 | EX_INT1 | IO Interrupt 1, asynchronous, low active, triggered on the falling edge. | I1E.4 | I1F.4 | I1SF4.0 |
| 3 | FH_ME | A synchronous 1-period wide pulse, indicating that mass erase of FLASH is done. When this ISR is serviced, FLASH is cleared. | I1E.5 | I1F.5 | I1SF5.0 |

Table 9-18 to Table 9-23 list all related registers to this interrupt.

**Table 9-18 Int1_n Interrupt Enable Register (I1E, 0x1C02)**

| 0x1C02, R/W | Int1_n Interrupt Enable Register, I1E | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | FME | EXI1N | - | - | PM1 | PM0 |
| Default Value | - | - | 0 | 0 | - | - | 0 | 0 |

- When EA=1 (IE.7), and EX1=1 (IE.2), all sub-interrupts of Int1_n interrupt can be enabled or disabled via configuring the register I1E.
- 1, to enable the interrupt; 0, to disable the interrupt.

**Table 9-19 Int1_n Interrupt Flag Register (I1F, 0x1C03)**

| 0x1C03, R | Int1_n Interrupt Flag Register, I1F | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | FME | EXI1N | - | - | PM1 | PM0 |
| Default Value | - | - | 0 | 0 | - | - | 0 | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

**Table 9-20 Active CF Pulse Interrupt Flag Register (I1SF0, 0x1C15)**

| 0x1C15, R/W | Active CF Pulse Interrupt Flag Register, I1SF0 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | PM0 |
| Default Value | - | - | - | - | - | - | - | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

**Table 9-21 Reactive Pulse CF Interrupt Flag Register (I1SF1, 0x1C16)**

| 0x1C16, R/W | Reactive Pulse CF Interrupt Flag Register, I1SF1 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | PM1 |
| Default Value | - | - | - | - | - | - | - | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

**Table 9-22 IO Interrupt 1 Flag Register (I1SF4, 0x1C19)**

| 0x1C19, R/W | IO Interrup 1 Flag Register, I1SF4 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | EXI1N |
| Default Value | - | - | - | - | - | - | - | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

**Table 9-23 Flash Mass Erase Flag Register (I1SF5, 0x1C1A)**

| 0x1C1A, R/W | Flash Mass Erase Flag Register, I1SF5 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | FME |
| Default Value | - | - | - | - | - | - | - | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

## 9.3.4. Int2

The interrupt Int2 is extended into 2 sub-interrupts, as shown in Table 9-24. Each sub-interrupts has its enable bit and interrupt flag, among which, the only-read register I2F can sign all interrupt flags, and the registers I2SF0 ~ I2SF1, which can be read and written, can set all sub-interrupt flag respectively. When interrupt flags are set to 1, and the corresponding interrupts are enabled, a 4-period wide pulse is sent periodically by the extended interrupt module on Int2 interrupt.

**Table 9-24 Int2 Interrupt**

| No. | Interrupt Source | Description | Enable Bit | Interrupt Flag (Read) | Interrupt Flag (R/W) |
|---|---|---|---|---|---|
| 0 | RTC_int | RTC second interrupt, an asynchronous signal. | I2E.0 | I2F.0 | I2SF0.0 |
| 1 | RTC_ille | RTC illegal interrupt, a synchronous 1-period wide pulse. | I2E.1 | I2F.1 | I2SF1.0 |

Table 9-25 to Table 9-28 list all related registers to this interrupt.

**Table 9-25 Int2 Interrupt Enable Register (I2E, 0x1C04)**

| 0x1C04, R/W | Int2 Interrupt Enable Register, I2E | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | ILIE | RTCIE |
| Default Value | - | - | - | - | - | - | 0 | 0 |

- When EA=1 (IE.7), and EX2=1 (EIE.0), all sub-interrupts of Int2 interrupt can be enabled or disabled via configuring the register I2E.
- 1, to enable the interrupt; 0, to disable the interrupt.

**Table 9-26 Int2 Interrupt Flag Register (I2F, 0x1C05)**

| 0x1C05, R | Int2 Interrupt Flag Register, I2F | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | ILIE | RTCIE |
| Default Value | - | - | - | - | - | - | 0 | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

**Table 9-27 RTC Periodic Interrupt Flag Register (I2SF0, 0x1C1C)**

| 0x1C1C, R/W | RTC Periodic Interrupt Flag Register, I2SF0 |
|---|---|

| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
|---|---|---|---|---|---|---|---|---|
| | - | - | - | - | - | - | - | RTCIF |
| Default Value | - | - | - | - | - | - | - | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

**Table 9-28 RTC Illegal Interrupt Flag Register (I2SF1, 0x1C1D)**

| 0x1C1D, R/W | RTC Illegal Interrupt Flag Register, I2SF1 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | ILIF |
| Default Value | - | - | - | - | - | - | - | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

## 9.3.5. Int3_n

The interrupt Int3_n is extended into 8 sub-interrupts, as shown in Table 9-29. They are the TimerA/B overflow interrupts and compare/capture interrupts. The enable bits and writable flags are embedded in Timer A/TimerB. The only-read register I3F can sign all interrupt flags. When interrupt flags are set to 1, and the corresponding interrupt is enabled, a 4-period wide pulse is sent periodically by the extended interrupt module on Int3_n interrupt.

When EA=1 (IE.7) and EX3=1 (EIE.1), TimerA/B overflow interrupts are enabled when the bits TAIE/TBIE (bit1 of TACTL and TBCTL, 0x1400/0x1500) are set, and TimerA/B compare/capture interrupts are enabled when the bits CCIE (bit4 of TACCTLx/TBCCTLx, x=0/1) are set.

**Table 9-29 Int3_n Interrupt**

| No. | Interrupt Source | Description | Interrupt Flag (Read) | Interrupt Flag (R/W) |
|---|---|---|---|---|
| 0 | TA_tai | Timer A Overflow Interrupt, triggered on high level | I3F.0 | TAIFG, TACTL.0 |
| 1 | TA_cci1 | Timer A Compare/Capture Interrupt 1, triggered on high level | I3F.2 | CCIFG, TACCTL1.0 |
| 2 | TA_cci2 | Timer A Compare/Capture Interrupt 2, triggered on high level | I3F.3 | CCIFG, TACCTL2.0 |
| 4 | TB_tai | Timer B Overflow Interrupt, triggered on high level | I3F.4 | TBIFG, TBCTL.0 |

| | | | | |
|---|---|---|---|---|
| 5 | TB_cci1 | Timer B Compare/ Capture Interrupt 1, triggered on high level | I3F.6 | CCIFG, TBCCTL1.0 |
| 6 | TB_cci2 | Timer B Compare/ Capture Interrupt 2, triggered on high level | I3F.7 | CCIFG, TBCCTL2.0 |

Table 9-30 lists the related register to this interrupt.

**Table 9-30 Int3_n Interrupt Flag Register (I3F, 0x1C07)**

| 0x1C07, R | Int3_n Interrupt Flag Register, I3F | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | TBC2 | TBC1 | - | TBT | TAC2 | TAC1 | - | TAT |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

## 9.3.6. Int4

The interrupt Int4 is extended into 2 sub-interrupt, as shown in Table 9-31. They are transmitting and receiving interrupts of UART2. The writable flags are embedded in UART. The only-read register I4F can sign all interrupt flags. When all flags are set to 1, and the corresponding interrupts are enabled, a 4-period wide pulse is sent periodically by the extended interrupt module on Int4_n interrupt.

When EA=1 (IE.7) and EX4=1 (EIE.2), the receiving and transmitting interrupt of UART2 are enabled.

**Table 9-31 Int4 Interrupt**

| No. | Interrupt Source | Description | Interrupt Flag (Read) | Interrupt Flag (R/W) |
|---|---|---|---|---|
| 0 | UART2_ri | Receiving Interrupt of UART2, triggered on the high level | I4F.0 | RI, SCON2.0 |
| 1 | UART2_ti | Transmitting Interrupt of UART2, triggered on the high level | I4F.1 | TI, SCON2.1 |

Table 9-32 lists the related registers to this interrupt.

**Table 9-32 Int4_n Interrupt Flag Register (I4F, 0x1C09)**

| 0x1C09, R | Int4_n Interrupt Flag Register, I4F |
|---|---|

| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
|---|---|---|---|---|---|---|---|---|
| | - | - | - | - | - | - | U2T | U2R |
| Default Value | - | - | - | - | - | - | 0 | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

## 9.3.7. Int5_n

The interrupt Int5_n is extended into 4 sub-interrupts, as shown in Table 9-33. They are transmitting and receiving interrupts of UART3 and UART5. The writable flags are embedded in UART. The only-read register I5F can sign all interrupt flags. When interrupt flags are set to 1, and the corresponding interrupts are enabled, a 4-period wide pulse is sent periodically by the extended interrupt module on Int5_n interrupt.

When EA=1 (IE.7) and EX5=1 (EIE.3), the transmitting and receiving interrupt of UART3 and UART5 are enabled.

**Table 9-33 Int5_n Interrupt**

| No. | Interrupt Source | Description | Interrupt Flag (Read) | Interrupt Flag (R/W) |
|---|---|---|---|---|
| 0 | UART3_ri | Receiving Interrupt of UART3, triggered on the high level | I5F.0 | SCON3.0 |
| 1 | UART3_ti | Transmitting Interrupt of UART3, triggered on the high level | I5F.1 | SCON3.1 |
| 2 | UART5_ri | Receiving Interrupt of UART5, triggered on the high level | I5F.2 | SCON5.0 |
| 3 | UART5_ti | Transmitting Interrupt of UART5, triggered on the high level | I5F.3 | SCON5.1 |

Table 9-34 lists the related register to this interrupt.

**Table 9-34 Int5_n Interrupt Flag Register (I5F, 0x1C0B)**

| 0x1C0B, R | Int5_n Interrupt Flag Register, I5F | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | U5T | U5R | U3T | U3R |
| Default Value | - | - | - | - | 0 | 0 | 0 | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

## 9.3.8. WDTI

As for V9003/V9103, the interrupt WDTI is extended into 7 sub-interrupts, as shown in Table 9-35. They are transmitting and receiving interrupts of UART4 and the P3 Port Interrupt. The enable bits and writable flags of the P3 port interrupt are embedded in GPIO, and the writable flags of the interrupts of UART4 are embedded in UART. The only-read register WIF can sign the interrupt flag of UART4, and the only-read register P3F can sign the interrupt flag of P3. When all flags are set to 1, and the corresponding interrupts are enabled, a 4-period wide pulse is sent periodically by the extended interrupt module on WDTI interrupt.

When EA=1 (IE.7) and EWDI=1 (EIE.4), the transmitting and receiving interrupt of UART4 are enabled. The P3 port interrupts are enabled by the register P3INTE (0x1A37).

**Table 9-35 WDTI Interrupt**

| No. | Interrupt Source | Description | Interrupt Flag (Read) | Interrupt Flag (R/W) |
|---|---|---|---|---|
| 0 | UART4_ri | Receiving Interrupt of UART4, triggered on the high level | WIF.0 | SCON4.0 |
| 1 | UART4_ti | Transmitting Interrupt of UART4, triggered on the high level | WIF.1 | SCON4.1 |
| 2 | P3.1 | IO Interrupt 1 of P3 | P3F.1 | P3IF1.0 |
| 3 | P3.2 | IO Interrupt 2 of P3 | P3F.2 | P3IF2.0 |
| 4 | P3.4 | IO Interrupt 4 of P3 | P3F.4 | P3IF4.0 |
| 5 | P3.5 | IO Interrupt 5 of P3 | P3F.5 | P3IF5.0 |
| 6 | P3.6 | IO Interrupt 6 of P3 | P3F.6 | P3IF6.0 |

Table 9-36 lists the related register to this interrupt.

**Table 9-36 WDT Interrupt Flag Register (WIF, 0x1C0D)**

| 0x1C0D, R | WDTI Interrupt Flag Register, WIF | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | U4T | U4R |
| Default Value | - | - | - | - | - | - | 0 | 0 |

1, the interrupt occurs; 0, no interrupt occurs.

# 9.4. Interrupt Resources

If two interrupts of the same tier (Interrupt Priority 1 or Interrupt Priority 0) occur simultaneously, the polling sequence of them is observed. The lower polling sequence number is, the higher polling priority the interrupt has. The interrupt PFI has the highest interrupt priority. Once an interrupt is being serviced, only an interrupt of higher priority can interrupt the service routine of the interrupt currently being serviced.

**Table 9-37 Interrupt Resources**

| Interrupt Polling Sequence | Interrupt | Vector | Interrupt No. | Description | Enables Bit in SFRs | Extended Enable Bits | Flag in SFRs | Extended Flag | Priority Control |
|---|---|---|---|---|---|---|---|---|---|
| 0 | PFI | 33H | 6 | IAP, Flash Erase/ Write of IAP | EICON.5 | | EICON.4 | | Highest |
| 1 | Int0_n | 03H | 0 | Powerup interrupt, I0SF0.0 | IE.0 | I0E.0 | | I0F.0 | IP.0 |
| | | | | Powerdown interrupt, I0SF1.0 | | I0E.1 | | I0F.1 | |
| | | | | PLL lock interrupt, I0SF2.0 | | I0E.2 | | I0F.2 | |
| | | | | IR interrupt, I0SF3.0 | | I0E.3 | | I0F.3 | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | IO interrupt 0, I0SF4.0 | | I0E.4 | | I0F.4 | |
| 2 | TF0 | 0BH | 1 | Timer 0 overflow interrupt | IE.1 | | TCON.5 | | IP.0 |
| 3 | Int1_n | 13H | 2 | Active CF pulse interrupt I1SF0.0 | IE.2 | I1E.0 | | I1F.0 | IP.2 |
| | | | | Reactive CF pulse interrupt I1SF1.0 | | I1E.1 | | I1F.1 | |
| | | | | IO interrupt 1 I1SF4.0 | | I1E.4 | | I1F.4 | |
| | | | | FLASH mass erase interrupt I1SF5.0 | | I1E.5 | | I1F.5 | |
| 4 | TF1 | 1Bh | 3 | Timer 1 overflow interrupt | IE.3 | | TCON.7 | | IP.3 |
| 5 | UART0 | 23h | 4 | Receiving interrupt of UART0, RI0 | IE.4 | | SCON0.0 | | IP.4 |
| | | | | Transmitting Interrupt of UART0, | | | SCON0.1 | | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | TI0 | | | | | |
| 6 | TF2 | 2Bh | 5 | Timer 2 overflow interrupt | IE.5 | | T2CON.7 | | IP.5 |
| 7 | UART1 | 3Bh | 7 | Receiving Interrupt of UART1, RI1 | IE.6 | | SCON1.0 | | IP.6 |
| | | | | Transmitting Interrupt of UART1, TI1 | | | SCON1.1 | | |
| 8 | Int2 | 43h | 8 | RTC second interrupt I2SF0.0 | EIE.0 | I2E.0 | EXIF.4 | I2F.0 | EIP.0 |
| | | | | RTC illegal interrupt I2SF1.0 | | I2E.1 | | I2F.1 | |
| 9 | Int3_n | 4Bh | 9 | Timer A Overflow Interrupt TAIFG | EIE.1 | TAIE | EXIF.5 | I3F.0 | EIP.1 |
| | | | | Reserved | | | | | |
| | | | | Timer A Compare/Capture Interrupt 1 CCIFG, TACCTL1.0 | | TACCTL1, CCIE | | I3F.2 | |
| | | | | Timer A | | TACCTL2, CCIE | | I3F.3 | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | Compare/Capture Interrupt2<br>CCIFG, TACCTL2.0 | | | | | |
| | | | | Timer B Overflow Interrupt<br>TBIFG | | TBIE | | I3F.4 | |
| | | | | Reserved | | | | | |
| | | | | Timer B Compare/Capture Interrupt 1<br>CCIFG, TBCCTL1.0 | | TBCCTL1, CCIE | | I3F.6 | |
| | | | | Timer B Compare/Capture Interrupt 2<br>CCIFG, TBCCTL2.0 | | TBCCTL2, CCIE | | I3F.7 | |
| 10 | Int4 | 53h | 10 | Receiving Interrupt of UART2<br>RI | EIE.2 | | EXIF.6 | I4F.0 | EIP.2 |
| | | | | Transmitting Interrupt of UART2<br>TI | | | | I4F.1 | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 11 | Int5_n | 5Bh | 11 | Receiving Interrupt of UART3<br>SCON3.0 | EIE.3 | | EXIF.7 | I5F.0 | EIP.3 |
| | | | | Transmitting Interrupt of UART3<br>SCON3.1 | | | | I5F.1 | |
| | | | | Receiving Interrupt of UART5<br>SCON5.0 | | | | I5F.2 | |
| | | | | Transmitting Interrupt of UART5<br>SCON5.1 | | | | I5F.3 | |
| 12 | WDTI | 63h | 12 | Receiving Interrupt of UART4<br>SCON4.0 | EIE.4 | | EICON.3 | WIF.0 | EIP.4 |
| | | | | Transmitting Interrupt of UART4<br>SCON4.1 | | | | WIF.1 | |
| | | | | IO Interrupt of GPIO (5) | | P3INTE.1/2/4/5/6 | | P3IF.1/2/4/5/6 | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | P3IF1/2/4/5/6.0 | | | | | |

# 10. Timers/Counters

There are 5 timers in V9003/V9103, including 3 timers (Timer0, Timer1 and Timer2) in 8052 and 2 16-bit timers (Timer A and Timer B) with similar functions of those in MSP430.

POR/BOR, RST Pin Reset, IO/RTC Wake-up Reset, WDT Reset or Debug Reset can set all of the timers/counters to the default state. In Sleep or Deep Sleep Mode, they stop working and get into Low-power-consumption Mode.

## 10.1. Timer A/Timer B

Timer A and Timer B have the same function. They are 16-bit timers/counters, and have 4 working modes. Each one of them has 2 compare/capture modules, and 3 configurable output units with 8 output modes.

Table 10-1 lists all related registers to Timer A and Timer B, and Timer A is taken as example to instruct the function of all registers.

**Table 10-1 Registers Related to Timer A and Timer B**

| Address | Description | |
|---|---|---|
| 0x1400 | TACTL, Timer A Control Register | R/W |
| 0x1402~0x1403 | TAR, Timer A Timer/Counter Register | R |
| 0x1404~0x1405 | TACCTL0~TACCTH0, Timer A Compare/Capture Control Register 0 | R/W |
| 0x1406~0x1407 | TACCTL1~TACCTH1, Timer A Compare/Capture Control Register 1 | R/W |
| 0x1408~0x1409 | TACCTL2~TACCTH2, Timer A Compare/Capture Control Register 2 | R/W |
| 0x140A~0x140B | TACCR0, Timer A Compare/Capture Register 0 | R/W |
| 0x140C~0x140D | TACCR1, Timer A Compare/Capture Register 1 | R/W |
| 0x140E~0x140F | TACCR2, Timer A Compare/Capture Register 2[1] | R/W |
| 0x1500 | TBCTL, Timer B Control Register | R/W |
| 0x1502~0x1503 | TBR, Timer B Timer/Counter Register | R |
| 0x1504~0x1505 | TBCCTL0~TBCCTH0, Timer B Compare/Capture Control Register 0 | R/W |

[1] TimerA/B is designed with 3 compare/capture modules, but in V9003/V9103, only Module1 and Module2 are in use.

| | | |
|---|---|---|
| 0x1506~0x1507 | TBCCTL1~TBCCTH1, Timer B Compare/Capture Control Register 1 | R/W |
| 0x1508~0x1509 | TBCCTL2~TBCCTH2, Timer B Compare/Capture Control Register 2 | R/W |
| 0x150A~0x150B | TBCCR0, Timer B Compare/Capture Register 0 | R/W |
| 0x150C~0x150D | TBCCR1, Timer B Compare/Capture Register 1 | R/W |
| 0x150E~0x150F | TBCCR2, Timer B Compare/Capture Register 2 | R/W |

**Table 10-2 Timer A Counter/Timer Register (TAR, 0x1402~0x1403)**

| 0x1402~0x1403, R | Timer A Timer/Counter Register, TAR |
|---|---|
| TAR | This register gives the value of the 16-bit timer, Timer A. It is read-only, but it can be reset by software. Additionally, an interrupt is generated when Timer A overflows. |
| Default Value | 0 |

**Table 10-3 Timer A Control Register (TACTL, 0x1400)**

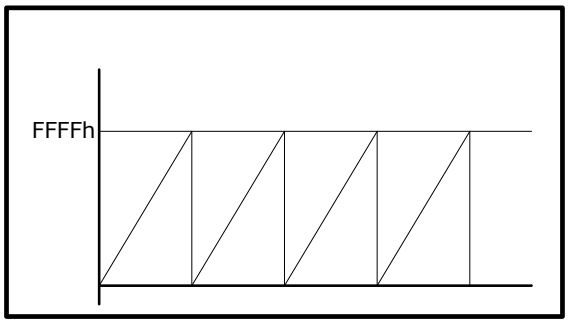
| 0x1400, R/W | Timer A Control Register, TACTL | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| TACTL | ID1 | ID0 | MC1 | MC0 | TSEL | CLR | TAIE | TAIFG |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 10-4 Bit Description of TACTL**

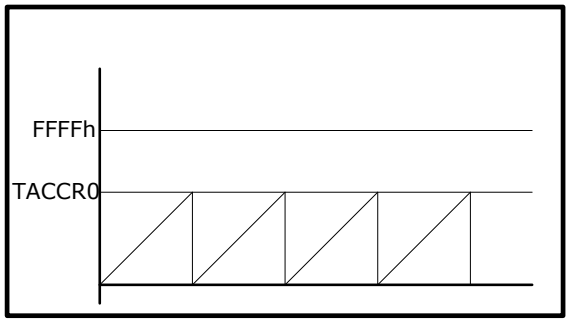
| Bit | Description |
|---|---|
| Bit7: Bit6 | ID1/ID0, to select the input divider. These bits select the divider for the input clock.<br>00, 2; 01, 4; 10, 8; 11, 16 |
| Bit5: Bit4 | MC1/MC0, to control the working mode, as shown in Figure 10-1.<br>00, stop mode: the timer is halted.<br>01, Up mode: the timer counts up to the value of TACCR0, and recount from 0000h;<br>10, Continuous mode: the timer counts up to FFFFh, and recounts from 0000h;<br>11, Up/down mode: the timer counts up to the value of TACCR0, and then, back down to 0000h. |
| Bit3 | TSEL, to select the clock source for the timer. |

| | |
|---|---|
| | 0, the timer use the value of ($f_{MCU}$/128) as the clock source;<br>1, the timer use $f_{MCU}$ as the clock source. |
| Bit2 | CLR, to clear the register TAR (0x1402~0x1403). When CLR=1, TAR is cleared; [ID1, ID0]=00; In Up/Down mode, the timer rolls over to 0000h, and back up to the value of TACCR0. |
| Bit1 | TAIE, TimerA Interrupt Enable. 1, to enable; 0, to disable.<br>When EX3=1 (EIE.1), and TAIE=1, TimerA overflow interrupt is enabled. |
| Bit0 | TAIFG, Timer A overflow interrupt flag;<br>Up Mode, when the timer rolls over to 0000h from the value of TACCR0, TAIFG=1;<br>Continuous Mode, when the timer rolls over to 0000h from FFFFh, TAIFG=1;<br>Up/Down Mode, when the timer counts down to 0000h from 0001h, TAIFG=1. |

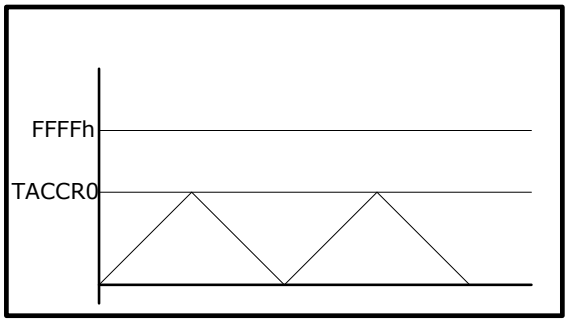
**Stop Mode**
Timer is stopped.




**Continuous Mode**
Up to FFFFh, rolls over to 0000, back up to FFFFh, etc.




**Up Mode**
Up to value specified by TACCR0, rolls over to 0000, back up to TACCR0 value, etc.




**Up/Down Mode**
Up to value specified by TACCR0, counts down to 0000h, back up to TACCR0 value, etc.

**Figure 10-1 Working Mode for TimerA/TimerB**

When MCx>0 and the clock source is active, the timer counts. Or, in Up or Up/Down Mode, when

0 is written into the register TACCR0, the timer may be stopped, and it may then be restarted incrementing in the up direction from zero when a nonzero value is written into the register TACCR0.

In Up Mode, when changing the value of the register TACCR0 while the timer is running, if the new value is greater than or equal to the former value, or greater than the current count value, the timer counts up to the new TACCR0; if the new value is less than the current count value, the timer rolls to 0000h, however, one additional count may occur before the counter rolls to 0000h.

In Up/Down Mode, when changing TACCR0 while the timer is running, and counting in the down direction, the timer continues its descent until it counts down to 0000h. The new value of TACCR0 takes affect after the timer counts down to 0000h. When the timer is counting in the up direction, and the new value is greater than or equal to the former value, or greater than the current count value, the timer counts up to the new TACCR0 value before counting down. When the timer is counting in the up direction, and the new value is less than the current count value, the timer begins counting down, however, one additional count may occur before the counter begins counting down.

In Continuous Mode, the output frequency is configurable, as shown in Figure 10-2.

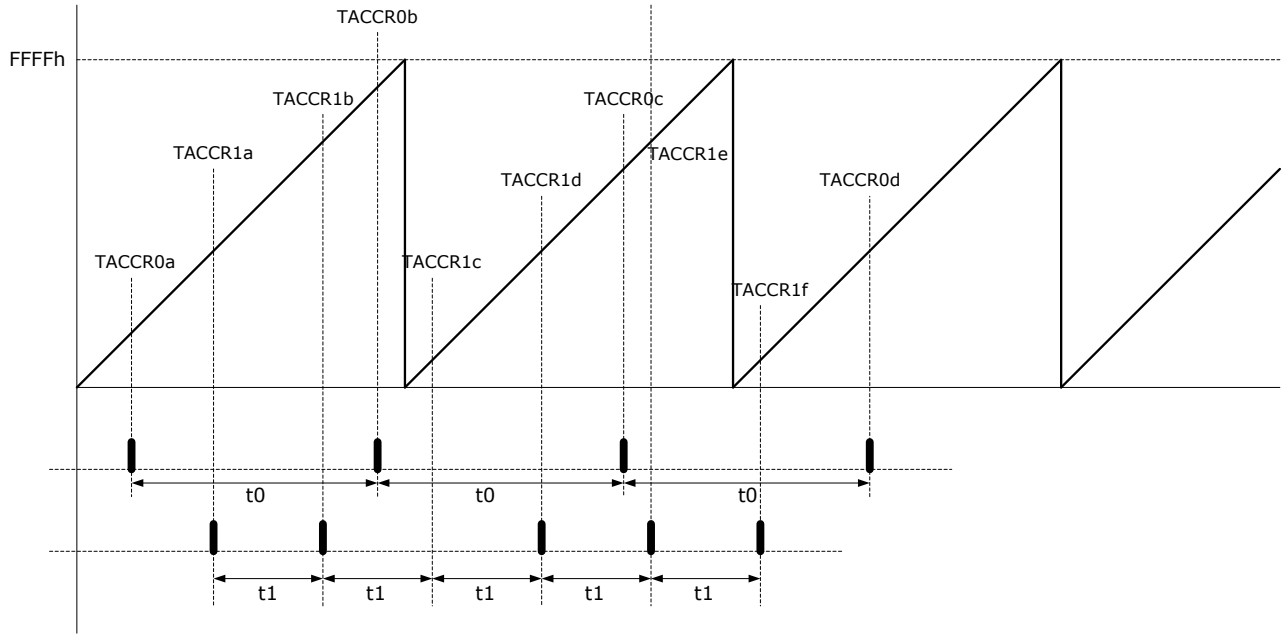



**Figure 10-2 Configuring the Output Frequency in Continuous Mode**

In Figure 10-2, TACCR0a or TACCR1a is the value of the register TACCR0 or TACCR1 at the moment of Ta0 or Ta1, and TACCR0b or TACCR1b is the value of the register TACCR0 or TACCR1 at the moment of Tb0 (Tb0=Ta0+t0) or Tb1 (Tb1=Ta1+t1), and so forth. The continuous mode can be used to generate independent output frequencies. When the interrupt is enabled (CCIE=1, Bit4, TACCTLx, Table 10-5), an interrupt is generated at an interval (t0 or t1). As shown in Figure 10-2, an interrupt is generated at the moment Ta0 and Ta1 independently, and the interrupt flags CCIFG (Bit0 of TACCTLx, x=1 and 2, Table 10-5) are set respectively. Up to 2 independent output

frequencies can be generated using all both capture/compare registers. In this usage, when the timer rolls over to 0000h from FFFFh, the bit TAIFG still is set.

**Table 10-5 Timer A Compare/Capture Control Registers (0x1406~0x1409)**

| 0x1406~0x1409, R/W | Time A Compare/ Capture Control Register x*, TACCTLx~TACCTHx | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9 | Bit8 |
| Higher Byte | CM1 | CM0 | CCIS1 | CCIS0 | SCS | SCCI | - | CAP |
| Default Value | - | - | - | - | - | - | 0 | 0 |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |
| Lower Byte | OUTMOD2 | OUTMOD1 | OUTMOD0 | CCIE | CCI | OUT | COV | CCIFG |
| Default Value | 0 | 0 | 0 | 0 | - | 0 | 0 | 0 |

* x=1~2, to indicate Timer A Capture/Compare Control Register 1/2, the addresses of which are listed in Table 10-1.

**Table 10-6 Bit Description of the Registers TACCTLx~TACCTLx**

<table>
<tr><th>Bit</th><th>Description</th></tr>
<tr><td>Bit[15:14]</td><td>CM1/CM0: Capture mode
<table>
<tr><th>Bit14~bit15</th><th>Capture Mode</th><th>Description</th></tr>
<tr><td>0</td><td>Disabled</td><td>To disable capture mode.</td></tr>
<tr><td>1</td><td>On the rising edge</td><td>To capture signals on the rising edge.</td></tr>
<tr><td>2</td><td>On the falling edge</td><td>To capture signals on the falling edge.</td></tr>
<tr><td>3</td><td>On both edges</td><td>To capture signals on both edges.</td></tr>
</table></td></tr>
<tr><td>Bit[13:12]</td><td>CCIS1/CCIS0: capture/compare input select. These bits select the input capture signal. In compare mode, both bits are unused. After reset, they are set to 0.<br>0, to select the input signal on the pins TA1/TA2;<br>1/2, 0;<br>3, 1.</td></tr>
<tr><td>Bit11</td><td>SCS: to synchronize the capture input signal with the timer clock. After reset, this bit is set to 0.<br>0, asynchronous capture;</td></tr>
</table>

<table>
<tr><td></td><td>1, synchronous capture.</td></tr>
<tr><td>Bit10</td><td>SCCI: to latch the selected input signal (determined by bits CCISx) via configuring the bits CCISx with the EQUx signal, and read it. After reset, this bit is read as 0.</td></tr>
<tr><td>Bit9</td><td>Read-only. Always read out as 0.</td></tr>
<tr><td>Bit8</td><td>CAP: to work in Capture or Compare mode.<br>0, Compare Mode, to generate PWM output signals or interrupts at specific timer intervals;<br>1, Capture Mode, to record time events, speed computation, or time measurement, to capture the value of the register TAR into the registers TACCR1/2.</td></tr>
<tr><td>Bit[7:5]</td><td>OUTMOD2/OUTMOD1/OUTMOD0: to select the output mode, see Figure 10-3 for pulse output.
<table>
<tr><th>Bit5~bit7</th><th>Output mode</th><th>Description</th></tr>
<tr><td>000</td><td>Output</td><td>To output the value of the bit OUT on the pin TAx (x=1~2).</td></tr>
<tr><td>001</td><td>Set</td><td>When TAR=TACCRx (x=1~2), the output on the corresponding pin TAx is set. It remains set until a reset of the timer, or until another output mode is selected and affects the output.</td></tr>
<tr><td>010</td><td>Toggle/Reset</td><td>When TAR=TACCRx (x=1~2), the output on the corresponding pin TAx is toggled;<br>When TAR=TACCR0, the output on the pin TAx is reset.<br>This mode is not for the output on the pin TA0.</td></tr>
<tr><td>011</td><td>Set/Reset</td><td>When TAR=TACCRx (x=1~2), the output on the corresponding pin TAx is set;<br>When TAR=TACCR0, the output on the pin TAx is reset.<br>This mode is not for the output on the pin TA0.</td></tr>
<tr><td>100</td><td>Toggle</td><td>When TAR= TACCRx (x=1~2), the output on the corresponding pin TAx is toggled.</td></tr>
<tr><td>101</td><td>Reset</td><td>When TAR= TACCRx (x=1~2), the output on the corresponding pin TAx is reset. It remains reset until another output mode is selected and affects the output.</td></tr>
<tr><td>110</td><td>Toggle/Set</td><td>When TAR=TACCRx (x=1~2), the output on the corresponding pin TAx is toggled;</td></tr>
</table></td></tr>
</table>

| | | | |
|---|---|---|---|
| | | | When TAR=TACCR0, the output on the pin TAx is set.<br>This mode is not for the output on the pin TA0. |
| | 111 | Reset/Set | When TAR=TACCRx (x=1~2), the output on the corresponding pin TAx is reset;<br>When TAR=TACCR0, the output on the pin TAx is set.<br>This mode is not for the output on the pin TA0. |
| Bit4 | CCIE: Interrupt enable bit, to enable TimerA compare/capture interrupt.<br>0, to disable; 1, to enable. | | |
| Bit3 | CCI: Compare/capture input signal<br>The selected input signal (via configuring the bits CCIS1/CCIS0) can be read out via this bit. After reset, this bit is set to 0. | | |
| Bit2 | When OUTMOD2/OUTMOD1/OUTMOD0=000, the value of the bit OUT is output on the pins TAx (x=1~2). | | |
| Bit1 | COV: Capture overflow flag.<br>In Capture mode,<br>When COV=0, the capture signal is reset, COV is not set;<br>When COV=1, if a capture occurs when the value of the last capture has not been read out, COV is set.<br>This bit must be reset by program. | | |
| Bit0 | CCIFG: compare / capture interrupt flag<br>In capture mode: CCIFG=1, the value of the register TAR is captured into the register TACCR1/2, see Figure 10-3 for details;<br>In compare mode: CCIFG=1, the value of the register TAR is equal to that of the register TACCR1/2 (EQUx signal), see Figure 10-3 for details.<br>In Compare/Capture Module 0, when the interrupt request is responded, CCIFG is reset automatically.<br>In Compare/Capture Module 1, when the interrupt request is responded, the CCIFG flag is reset; if the enable bit is cleared, no interrupt occurs, but the CCIFG flag still is set. So, the CCIFG flag must be reset by program. | | |

As shown in Figure 10-3, users can configure the bits OUTMOD2/OUTMOD1/OUTMOD0 to select the output mode. When these bits are set to 010/011/110/111, the frequency and duty cycle of the output pulse changes, generating PWM signals (pulse width modulation).

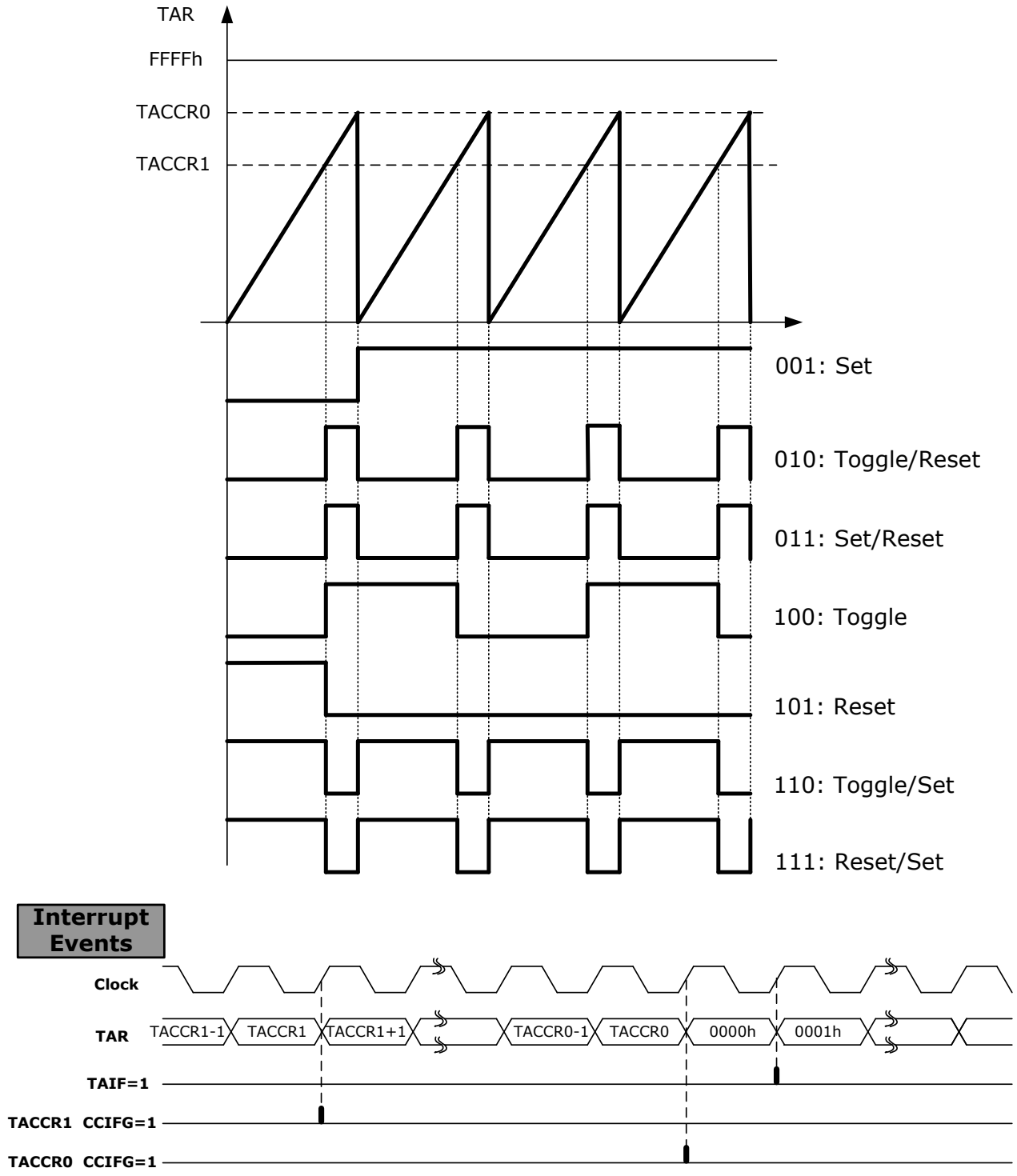



**Figure 10-3 Output on Pin TA1 in Up Mode**

# 10.2. Timer0/Timer1/Timer2

Besides TimerA and TimerB, there are 3 timers in V9003/V9103: Timer0, Timer1, and Timer2. Of them, Timer0 is a general timer, but Timer 1 and/or Timer2 can be both general timers and baud rate generators of UART0 and/or UART1.

Users can configure the register CKCON SFR to select the clock source (clk/12 or clk/4) for Timer0/Timer1/Timer2.

## 10.2.1.Timer Rate Control

When the bits, CKCON.5, CKCON.4 and CKCON.3, in CKCON SFR (0x8E), are set, the associated counters increment by ones every clock cycle (clk). When they are cleared, the associated counters increment by ones every 12 clock cycles (clk/12). The timer control is independent of each other. By default, these three bits are set to 0, that is, incrementing at twelve-clock intervals. These bits have no effect on counter mode.

**Table 10-7 CKCON Register Bits**

| Bit | Description |
|---|---|
| CKCON.5 | T2M – to select clock source for Timer 2. T2M=0, Timer 2 uses clk/12; T2M=1, Timer 2 uses clk/4. When Timer2 is configured for baud rate generation, Timer2 is forced to use clk/2 regardless of the configuration of this bit. |
| CKCON.4 | T1M – to select clock source for Timer 1. T1M=0, Timer 1 uses clk/12; T1M=1, Timer 1 uses clk/4. |
| CKCON.3 | T0M – to select clock source for Timer 0. T0M=0, Timer 0 uses clk/12; T0M=1, Timer 0 uses clk/4. |

## 10.2.2.Timer0, Timer1

Timer0 and Timer1 are two of three embedded timers of 8052 microcontroller. In V9003/V9103, both of them act as a timer to count the MCU clock frequency. Besides those, Timer1 also can act as a baudrate generator for serial communication.

There are 4 operation modes for Timer 0 and Timer1. They are determined by the TMOD SFR (0x89) and TCON SFR (0x88). The four modes are:

- 13-bit timer (Mode 0)
- 16-bit timer (Mode 1)
- 8-bit timer in auto-reload mode (Mode 2)
- Split 8-bit timer mode (Mode 3, only for Timer 0)

The SFRs associated with Timer0/Timer1 are:

- TMOD SFR (0x89) – Timer0/1 Mode Control Special Function Register
- TCON SFR (0x88) – Timer0/1 Control Special Function Register
- TL0 SFR (0x8A) and TH0 SFR (0x8C) – the lower byte and higher byte of Timer 0
- TL1 SFR (0x8B) and TH1 SFR (0x8D) – the lower byte and higher byte of Timer1

**Table 10-8 Timer0/1 Mode Control Special Function Register, TMOD SFR (0x89)**

| Bit | Description | | | |
|---|---|---|---|---|
| TMOD.7 | GATE1 – Timer1 gate control bit. If the bit TR1 is set and the signal on the pin INT1 (P2.5) is high, Timer1 runs when GATE1=1. If GATE1=0, Timer1 runs when TR1=1, regardless of the state of the pin INT1. | | | |
| TMOD.6 | C/T1 – Counter or counter select bit, only set to 0. When C/T1 = 0, Timer1 acts as a timer to count the clock frequency (clk/4 or clk/12, depending on the bit T1M, CKCON.4). | | | |
| TMOD.5 | T1M1 – Timer 1 Mode Select Bit 1 | T1M1 | T1M0 | Mode |
| | | 0 | 0 | Mode 0, 13-bit timer |
| TMOD.4 | T1M0 –Timer 1 Mode Select Bit 0 | 0 | 1 | Mode 1, 16-bit timer |
| | | 1 | 0 | Mode 2, 8-bit timer in auto-reload mode |
| | | 1 | 1 | Mode 3, Split 8-bit timer |
| TMOD.3 | GATE0 – Timer0 gate control bit. If the bit TR0 is set and the signal on the pin INT0 (P2.4) is high, Timer0 runs when GATE0=1. If GATE0=0, Timer0 runs when TR0=1, regardless of the state of the pin INT0. | | | |
| TMOD.2 | C/T0 – Counter or counter select bit, only set to 0. When C/T0 = 0, Timer0 acts as a timer to count the clock frequency (clk/4 or clk/12, depending on the bit T0M, CKCON.3). | | | |
| TMOD.1 | T0M1 – Timer 0 Mode Select Bit 1 | T0M1 | T0M0 | Mode |
| | | 0 | 0 | Mode 0, 13-bit timer |
| TMOD.0 | T0M0 –Timer 0 Mode Select Bit 0 | 0 | 1 | Mode 1, 16-bit timer |
| | | 1 | 0 | Mode 2, 8-bit timer in auto-reload mode |
| | | 1 | 1 | Mode 3, Split 8-bit timer |

**Table 10-9 Timer0/1 Control Special Function Register, TCON SFR (0x88)**

| Bit | Description |
|---|---|
| TCON.7 | TF1 – Timer1 overflow flag. It is set when Timer1 rolls from all ones to zeros. It is cleared when the processor vectors to execute interrupt service routine located at program address 001Bh ("Interrupt Resources"). |
| TCON.6 | TR1 –Timer 1 run control bit. Set the bit to turn on Timer 1. |
| TCON.5 | TF0 – Timer0 overflow flag. It is set when Timer0 rolls from all ones to zeros. It is cleared when the processor vectors to execute interrupt service routine located at program address 000Bh ("Interrupt Resources"). |
| TCON.4 | TR0 –Timer 0 run control bit. Set the bit to turn on Timer 0. |
| TCON.3 | IE1 – Reserved. |
| TCON.2 | IT1 – IO Interrupt 1 type select bit, must be 1. |
| TCON.1 | IE0 – Reserved. |
| TCON.0 | IT0 – IO Interrupt 0 type select bit, must be 1. |

## 10.2.2.1. Timer0/1, Mode0

In Mode 0, Timer0 and Timer1 act as13-bit timers. In this mode, the lower byte of Timer0/Timer1 (TLx SFR, 0x8A or 0x8B) counts from 0~31. When it increments from 31, TLx SFR (x=0~1) is cleared, and the higher byte of the timer (THx SFR, 0x8C or 0x8D) increments by 1. In this mode, only 13 bits of Timer0/Timer1, Bit0~Bit4 of TLx SFR and all 8 bits of THx SFR, are active. The upper three bits of TLx SFR are indeterminate in Mode 0 and must be masked when the software evaluates the register.

Users can configure the bits (TR0 or TR1, Bit4 or Bit6 of TCON SFR) to turn on Timer0 or Timer1. In V9003/V9103, according to the value of the bit C/Tx (x=0~1, Bit6 or Bit2 of TMOD SFR), Timer0 or Timer1 can act as a timer only.

When GATEx=0 (Bit7 or Bit3 of TMOD SFR), or GATEx=1and the input signal on the pin INT0 or INT1 is active, Timer0 or Timer1 runs when TRx (x=0~1, TCON.4 or TCON.6) is set.

When the 13-bit timer increments from 1FFFh, it rolls over to all zeros, and then, the bit TF0 (TCON.5) or TF1 (TCON.7) is set, and an interrupt is generated to CPU.

## 10.2.2.2. Timer 0/1, Mode 1

In Mode 1, Timer0 and Timer1 act as 16-bit timers. In this mode, all eight bits of the lower byte

of the timer, TL0 SFR (0x8A) or TL1 SFR (0x8B), are used, so, TLx SFR increments from 0 to 255. When TLx SFR increments from 255, it is cleared, and the higher byte of the timer, THx SFR (TH0 SFR or TH1 SFR), increments by 1. The timer will roll over to all zeros when the timer/counter increments from FFFFh.

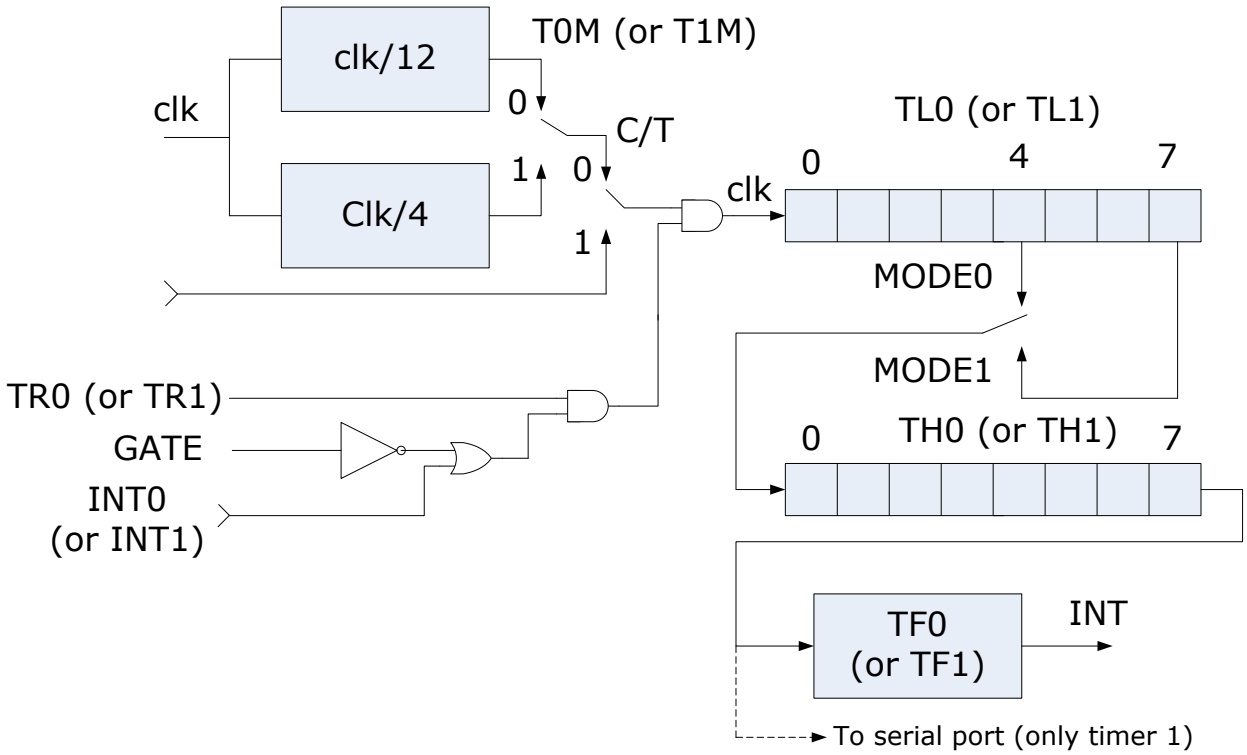



**Figure 10-4 Timer 0/1, Mode 0/1**

## 10.2.2.3. TImer 0/1, Mode 2

In Mode 2, only the lower byte of Timer0/Timer1 (TLx SFR, x=0~1) acts as an 8-bit timer, while the higher byte of it (THx SFR, x=0~1) holds a value that is loaded into TLx SFR every time TLx SFR overflows. When the value is loaded into the TLx SFR, it will increment from the loaded value.

For example, TH1 SFR is set to 200, and when TL1 SFR increments from 255, it rolls to 200, and recounts from 200 to 255, and then to 200, and repeats.

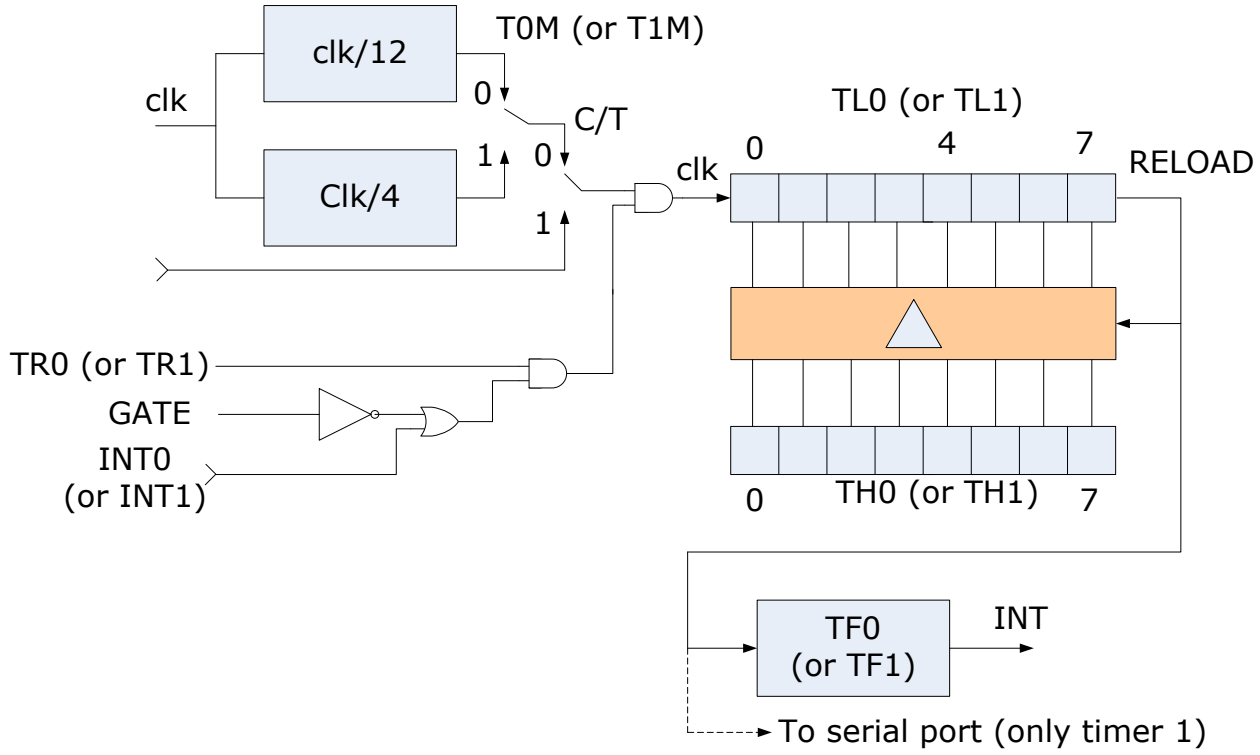



**Figure 10-5 Timer 0/1, Mode 2**

### 10.2.2.4. Timer 0/1, Mode 3

In Mode 3, Timer0 becomes two completely separate 8-bit timers. When Timer0 is set to work in this mode, TR0 (TCON.4) and TF0 (TCON.5) are used by TL0 SFR, but TR1 (TCON.6) and TF1 (TCON.7) are used by TH0 SFR, as shown in Table 10-10.

**Table 10-10 Timer0 in Mode 3**

| Timer | Run control bit | Overflow Interrupt Flag |
|---|---|---|
| TL0 SFR | TR0 (TCON.4). TR0=1, to enable the lower byte of Timer0 to count. | TF0 (TCON.5). It is set when TL0 SFR rolls from all ones to zeros. It is cleared when the processor vectors to execute interrupt service routine. |
| TH0 SFR | TR1 (TCON.6). TR1=1, to enable the higher byte of Timer0 to count. | TF1 (TCON.7). It is set when TH0 SFR rolls from all ones to zeros. It is cleared when the processor vectors to execute interrupt service routine. |

When Timer0 works in Mode3, Timer1 still can be configured to work in Mode0/1/2, but no interrupt will be generated by it, because the flag TF1 is used by Timer0.

When Timer1 is configured to work in Mode 3, it stops working, but holds its counts.

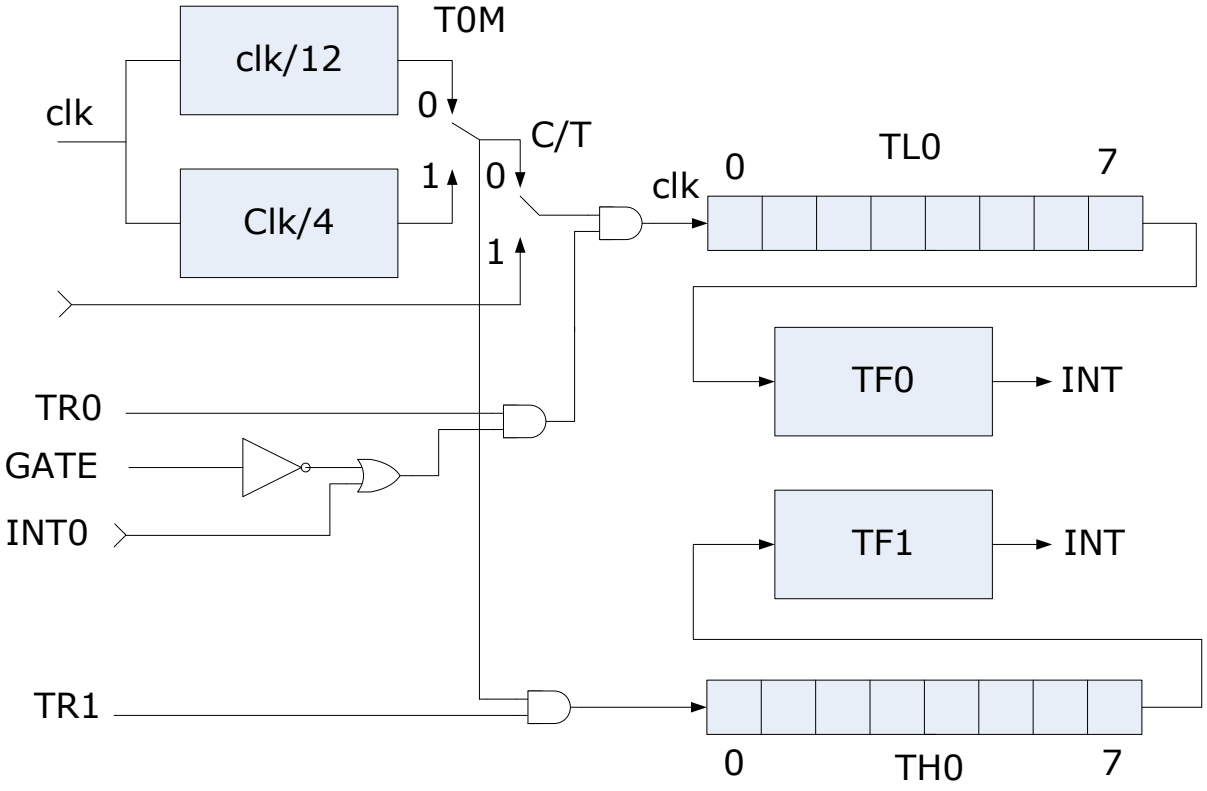



**Figure 10-6 Timer 0/1, Mode 3**

**Table 10-11 Configuring TMOD.5 and TMOD.4**

| TMOD.5 (T1M1) | TMOD.4 (T1M0) | Mode | Timer 1 |
|---|---|---|---|
| 0 | 0 | Mode 0 | On |
| 0 | 1 | Mode 1 | |
| 1 | 0 | Mode 2 | |
| 1 | 1 | Mode 3 | Off |

## 10.2.3.Timer2

Besides Timer0 and Timer1, there is a third timer, Timer2, in 8052 microcontroller, a 16-bit timer, capable of a number of new operation modes. The modes for Timer2 are:

- 16-bit timer
- 16-bit timer in auto-reload mode
- Baud rate generator for UART0

The SFRs associated with Timer 2 are:

- T2CON SFR (0xC8) – Timer2 Control Special Function Register
- TL2 SFR (0xCC) – Lower byte of Timer2
- TH2 SFR (0xCD) – Higher byte of Timer2
- RCAP2L SFR (0xCA) – To hold the lower byte of the load value of TL2 SFR when Timer2 is configured in auto-reload mode
- RCAP2H SFR (0xCB) – To hold the higher byte of the load value of TH2 SFR when Timer2 is configured in auto-reload mode

**Table 10-12 Timer2 Control Special Function Register, T2CON SFR (0xC8)**

| Bit | Description |
|---|---|
| T2CON.7 | TF2 –Timer 2 overflow flag. The bit TF2 is set when Timer 2 overflows from FFFFh. TF2 must be cleared by the program. Only when RCLK and TCLK are both cleared, TF2 is set. Writing 1 to TF2 forces a Timer 2 interrupt if it is enabled. |
| T2CON.6 | EXF2 – reserved |
| T2CON.5 | RCLK – Receive clock flag. RCLK=1, Timer2 overflow is used to determine receiver baud rate for UART0 interface.<br>When this bit is set, Timer2 is forced into baud rate generation mode. The timer will operate from a divide by 2 of the external clock. |
| T2CON.4 | TCLK – Transmit clock flag. TCLK=1, Timer2 overflow is used to determine transmitter baud rate for UART0 interface.<br>When this bit is set, Timer2 is forced into baud rate generation mode. The timer will operate from a divide by 2 of the external clock. |
| T2CON.3 | EXEN2 – Reload control bit. When EXEN2=1, auto-reload occurs when Timer 2 overflows. |
| T2CON.2 | TR2 –Timer 2 run control bit. TR2=1, Timer 2 is turned on; TR2=0, Timer 2 is turned off. |
| T2CON.1 | C/T2 – Counter or counter select bit, only set to 0. When C/T2 = 0, Timer2 acts as a timer to count the clock frequency (clk/4 or clk/12, depending on the bit T2M, CKCON.5).<br>When Timer2 is forced into baud rate generation mode, C/T2=0 enables Timer2 using clk/2 as the clock source, regardless of the configuration of the bit T2M (CKCON.5). |
| T2CON.0 | CP/RL2 – Reload flag, set to 0 only. If CP/RL2 = 0, and EXEN=2, auto-reloads occur when Timer 2 overflows. If either RCLK or TCLK is set to 1, CP/RL2 will not function, |

| | |
|---|---|
| | and Timer 2 will operate in auto-reload mode after overflows. CP/RL2 is set to 0 only.<br>When CP/RL2 = 0, and EXEN2=1, an auto-reload event occurs when Timer2 overflows. If either RCLK or TCLK is set, CP/RL2 cannot function, and Timer2 can operate in auto-reload mode every overflow. |

The following table summarizes how the bits of the T2CON SFR determine the Timer 2 mode.

**Table 10-13 Timer 2 Mode**

| RCLK | TCLK | CP/RL2 | TR2 | Mode |
|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 16-bit timer in auto-reload mode |
| 1 | X | X | 1 | Baud rate generator for data receive of UART0 |
| X | 1 | X | 1 | Baud rate generator for data transmission of UART0 |
| X | X | X | 0 | Off |

## 10.2.3.1. Timer2, 16-Bit Timer

In this mode, users can configure the register T2CON SFR to enable Timer2 acting as a 16-bit timer, and to enable Timer2 running (TR2, T2CON.2). In this mode, Timer2 increments from 0000h to FFFFh, and then, rolls over to all zeros, with setting the flag TF2 (T2CON.7) which generating an interrupt to the processor.

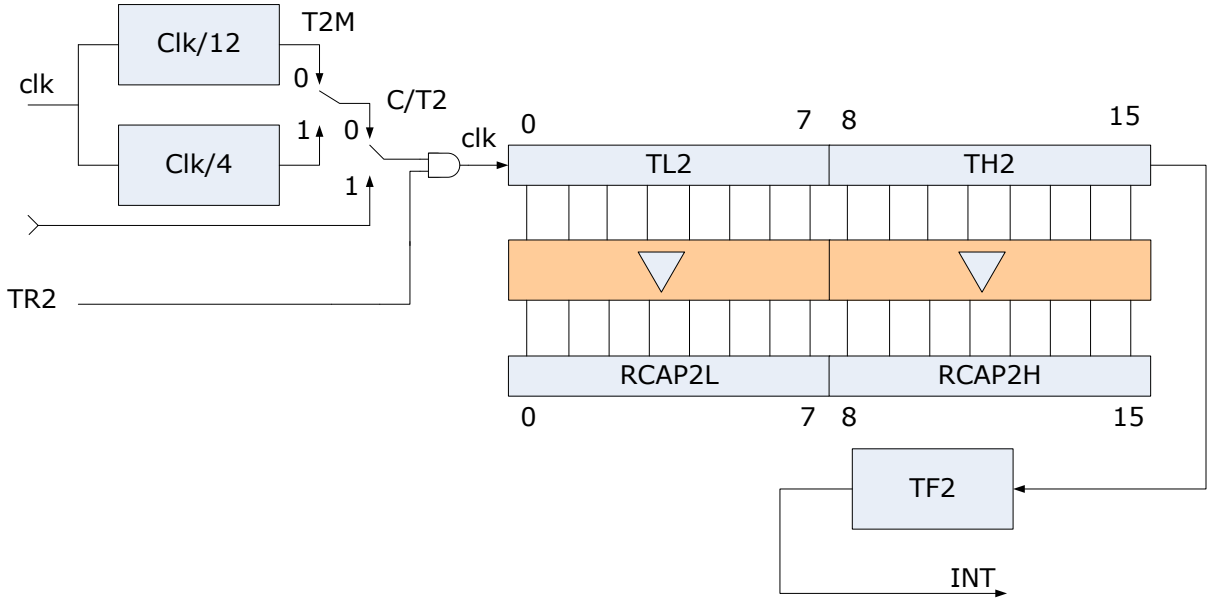



**Figure 10-7 Timer2, 16-bit Timer**

## 10.2.3.2. Timer2, 16-Bit Timer in Auto-Reload Mode

When CP/RL2 = 0 (T2CON.0), Timer 2 acts as a 16-bit timer in auto-reload mode.

In this mode, users must write the reload value into the registers RCAP2L SFR (0xCA) and RCAP2H SFR (0xCB).

When the timer increments from FFFFh, the value stored in RCAP2L SFR is reloaded into the register TL2 SFR (0xCC), and the value stored in RCAP2H SFR is reloaded into the register TH2 SFR (0xCD), at the same time, TF2 is set, which will generate an interrupt to the processor if it is enabled.

Note: The initial reloaded values must be configured in the registers RCAP2L SFR (0xCA) and RCAP2H SFR (0xCB) in advance.

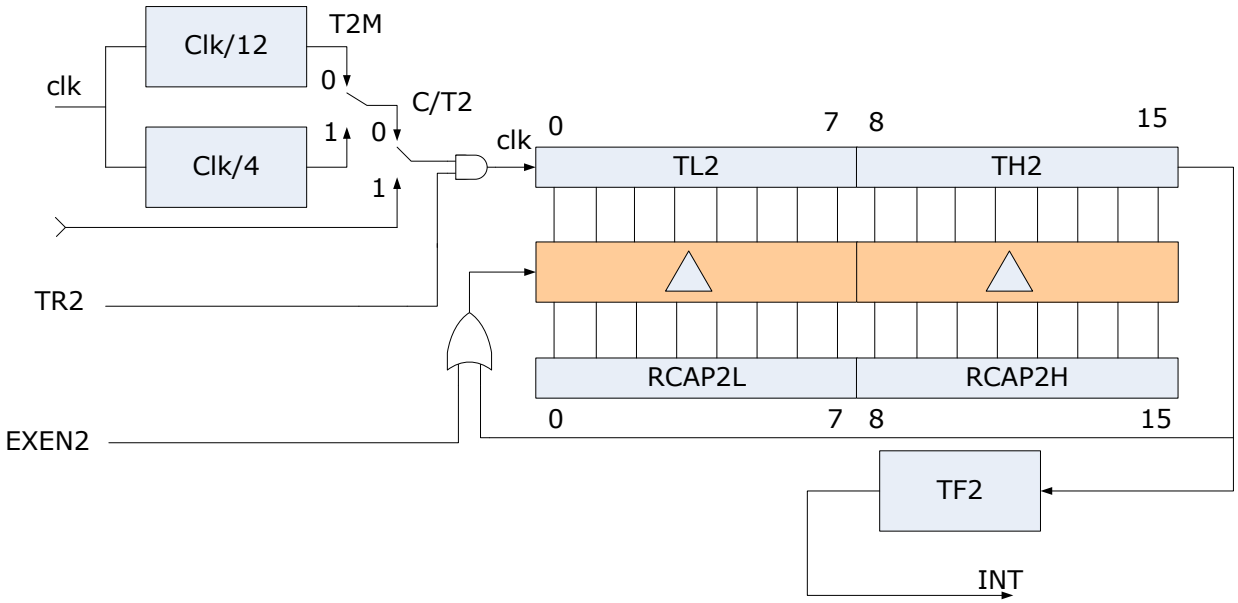



**Figure 10-8 Timer2, 16-Bit Timer in Auto-Reload Mode**

## 10.2.3.3. Baud Rate Generator Mode

When either TCLK (T2CON.4) or RCLK (T2CON.5) is set, Timer2 is forced into baud rate generation mode for UART0. In this mode, Timer2 is configured in auto-reload mode regardless of the configuration of the bit CP/RL2.

When Timer2 is used as the baud rate generator for UART0, the baud rate cannot be doubled even though the bit SMOD0 (PCON.7, 0x87) is set.

In the baud rate generation mode, C/T2=0 will force Timer2 to use clk/2 as the clock source; and TF2=0 cannot be set by hardware on a timer rollover, so Timer2 overflow interrupt is automatically disabled.

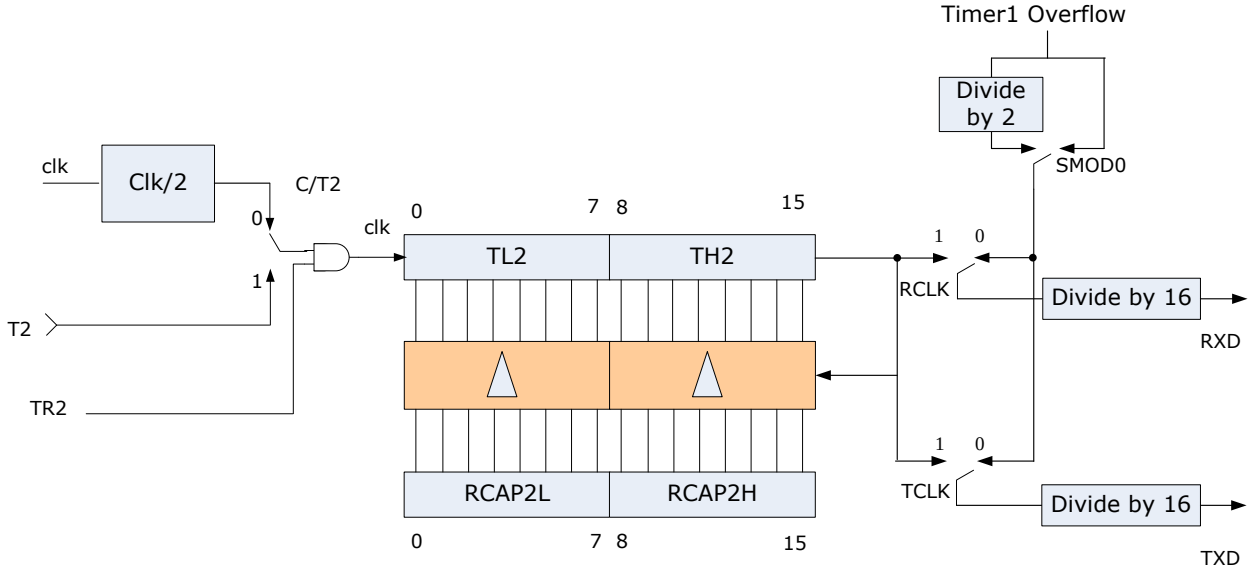



**Figure 10-9 Timer2, Baud Rate Generator Mode**

# 11. UART

There are total six UART serial interfaces in V9003/V9103, including embedded UART0 and UART1 in 8052, and additional UART2~UART5. The FIFO (first in first out) UART can be used to store 1-byte data.

When used for serial communication, UART0 and UART1 use Timer1 or/and Timer2 as the baud rate generators and UART2~UART5 use the embedded baud rate generator in them. Of them, only UART2 can support IR communication, transmitting optional 38kHz carrier wave.

The UART serial interfaces can work in 4 modes. In Mode 0, UART0 ~ UART5 can only receive data on the RXD port and output shifting clock on the TXD port. In other modes, the additional UART serial interfaces, UART2~UART5, can work like UART0 and UART1 serial interfaces of 8052 microcontroller.

It is recommended that UART2 ~ UART5 serial interfaces should be used for serial communication.

POR/BOR, RST Pin Reset, IO/RTC Wake-up Reset, WDT Reset or Debug Reset can set all of the UARTs to the default state. In Sleep or Deep Sleep Mode, they stop working and get into Low-power-consumption Mode.

## 11.1. UART0/ UART1

UART0 and UART1 use Timer 1 or/and Timer2 as the baud rate generator, and cannot transmit 38kHz carrier wave on their TXD ports.

### 11.1.1.UART0

The bit SMOD0 (PCON.7) in the register PCON SFR (0x87) controls whether the baud rate of UART0 is doubled or not. Both Timer1 and Timer2 can be used as the baud rate generator for UART0. When Timer1 is used, SMOD0=1 enables the baud rate being doubled; when Timer2 is used, SMOD0 has no effect on the baud rate of UART0.

The SFRs associated with the UART0 are:

SCON0 SFR (0x98) – UART0 Control Special Function Register

SBUF0 SFR (0x99) – UART0 Buffer Special Function Register

**Table 11-1 UART0 Control Special Function Register, SCON0 SFR (0x98)**

| Bit | Description | |
|---|---|---|
| SCON0.7 | SM00 – UART0 mode bit 0 | SM00 SM10 Mode<br>0 0 0: 8-bit shift register, baud rate=clk/4 or clk/12<br>0 1 1: 8-bit UART, baud rate is determined by Timer1 and/or Timer2<br>1 0 2: 9-bit UART, baud rate= clk/32 or clk/64<br>1 1 3: 9-bit UART, baud rate is determined by Timer1 and/or Timer2 |
| SCON0.6 | SM10 – UART0 mode bit 1 | |
| SCON0.5 | SM20 – Multiprocessor communication enable bit. In Mode 2 and 3, SM20 enables the multiprocessor communication. In Mode 2 or 3, when SM20=1, RI0 cannot be set in case that the received 9th bit is 0. In Mode 1, when SM20=1, RI0 can be set only if a valid stop bit is received. In Mode 0, SM20 establishes the baud rate: when SM20 = 0, the baud rate is clk/12; when SM20 = 1, the baud rate is clk/4. | |
| SCON0.4 | REN0 – Receive enable bit. When REN0 = 1, data receiving is enabled. | |
| SCON0.3 | TB80 – To define the 9th bit transmitted in Mode 2 and 3. | |
| SCON0.2 | RB80 – In Mode 2 and 3, RB80 indicates whether or not to store the received 9th bit. In Mode 1, RB80 indicates the received stop bit is stored. In Mode 0, RB80 is not used. | |
| SCON0.1 | TI0 – Transmit interrupt flag, indicating that data transmit data has been shifted out. In Mode 0, TI0 is set at the end of the 8th bit. In all other modes, TI0 is set when the stop bit is placed on the pin TXD0. TI0 must be cleared by the program. | |
| SCON0.0 | RI0 – Receive interrupt flag, indicating that a serial of data have been received. In Mode 0, RI0 is set after receiving the 8th data bit. In Mode 1, according to the state of SM20, RI0 is set after the last sample of the incoming stop bit. In Mode 2 and 3, RI0 is set at the end of the last sample of the 9th bit. RI0 must be cleared by the program. | |

## 11.1.2.UART1

The bit SMOD1 (EICON.7) in the register EICON SFR (0xD8) controls whether the baud rate of UART1 is doubled or not. Only Timer1 can be used as the baud rate generator for UART1.

The SFRs associated with the UART1 are:

SCON1 SFR (0xC0) – UART1 Control Special Function Register

SBUF1 SFR (0xC1) – UART1 Buffer Special Function Register

**Table 11-2 UART1 Control Special Function Register, SCON1 SFR (0xC0)**

| Bit | Description | | |
|---|---|---|---|
| SCON1.7 | SM01 – UART1 mode bit 0 | SM01 | SM11 Mode |
| | | 0 | 0 0: 8-bit shift register, baud rate=clk/4 or clk/12 |
| SCON1.6 | SM11 – UART1 mode bit 1 | 0 | 1 1: 8-bit UART, baud rate is determined by Timer1 |
| | | 1 | 0 2: 9-bit UART, baud rate= clk/32 or clk/64 |
| | | 1 | 1 3: 9-bit UART, baud rate is determined by Timer1 |
| SCON1.5 | SM21 – Multiprocessor communication enable bit. In Mode 2 and 3, SM21 enables the multiprocessor communication. In Mode 2 or 3, when SM21=1, RI1 cannot be set in case that the received 9th bit is 0. In Mode 1, when SM21=1, RI1 can be set only if a valid stop bit is received. In Mode 0, SM21 establishes the baud rate: when SM21 = 0, the baud rate is clk/12; when SM21 = 1, the baud rate is clk/4. | | |
| SCON1.4 | REN1 – Receive enable bit. When REN1 = 1, data receiving is enabled. | | |
| SCON1.3 | TB81 – To define the 9th bit transmitted in Mode 2 and 3. | | |
| SCON1.2 | RB81 – In Mode 2 and 3, RB81 indicates whether or not to store the received 9th bit. In Mode 1, RB81 indicates the received stop bit is stored. In Mode 0, RB81 is not used. | | |
| SCON1.1 | TI1 – Transmit interrupt flag, indicating that data transmit data has been shifted out. In Mode 0, TI1 is set at the end of the 8th bit. In all other modes, TI1 is set when the stop bit is placed on the pin TXD1. TI1 must be cleared by the program. | | |
| SCON1.0 | RI1 – Receive interrupt flag, indicating that a serial of data have been received. In Mode 0, RI1 is set after receiving the 8th data bit. In Mode 1, according to the state of SM21, RI1 is set after the last sample of the incoming stop bit. In Mode 2 and 3, RI1 is set at the end of the last sample of the 9th bit. RI1 must be cleared by the program. | | |

# 11.2. UART2

UART2 can be configured to transmit optional 38kHz carrier wave, of which the duty cycle is configurable, see "IR Modulation/Demodulation" for details.

There is an embedded baud rate generator (compatible with Timer1) in UART2. As an additional peripheral, there is a dedicated control/status register that controls the baud rate of UART2, clock source, on/off status, and overflow status of the baud rate generator, etc.

The UART2 related registers are listed in Table 11-3.

**Table 11-3 Registers Related to UART2**

| | |
|---|---|
| 0x1600, R/W | TCON2, UART2 Control / Status Register |
| 0x1601, R/W | TMOD2, UART2 Baud Rate Generator Mode Control Register |
| 0x1603, R/W | TH21, Higher Byte of Baud Rate Generator of UART2 |
| 0x1605, R/W | TL21, Lower Byte of Baud Rate Generator of UART2 |
| 0x1606, R/W | SCON2, UART2 Control Register |
| 0x1607, R/W | SBUF2, UART2 Buffer Register |

**Table 11-4 UART2 Control/ Status Register (TCON2, 0x1600)**

| 0x1600, R/W | UART2 Control/ Status Register, TCON2 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | SMOD | - | T1M | - | TF1 | - | TR1 | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- SMOD: double UART2 baud rate enable bit. SMOD=1, UART2 baud rate doubles;
- T1M: to select the clock source for baud rate generator; 0, to use clk/12; 1, to use clk;
- TF1: baud rate generator overflow flag, no influence on the overflow interrupt. 1, overflow; 0, non-overflow.
- TR1: baud rate generator run control bit. 1, to run; 0, to stop.

**Table 11-5 UART2 Baud Rate Generator Mode Control Register, TMOD2, 0x1601**

| 0x1601, R/W | UART2 Baud Rate Generator Mode Control Register, TMOD2 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | GATE | C/T | M1 | M0 | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- GATE: Gate control bit of the baud rate generator, set to 0 only.
- C/T: Timer/Counter select bit for the baud rate generator, set to 0 only.
- M1/M2: Baud rate generator operation mode select bit, see Table 11-6 for details.

**Table 11-6 Baud Rate Generator Operation Mode Select for UART2**

| M1 | M0 | Mode |
|---|---|---|
| 0 | 0 | Mode 0: 13-bit timer |
| 0 | 1 | Mode 1: 16-bit timer |
| 1 | 0 | Mode 2: 8-bit timer in auto-reload mode |
| 1 | 1 | Mode 3: Split 8-bit timer |

- Generally, the baud rate generator works in Mode 2: 8-bit timer in auto-reload mode.

**Table 11-7 Bit Description of SCON2**

| Bit | Description | |
|---|---|---|
| SCON2.7 | SM0 – UART2 mode bit 0 | SM0 SM1 Mode<br>0 0 0: 8-bit shift register, baud rate=clk or clk/12<br>0 1 1: 8-bit UART, baud rate is determined by the baud rate generator<br>1 0 2: 9-bit UART, baud rate= clk/32 or clk/64<br>1 1 3: 9-bit UART, baud rate is determined by the baud rate generator |
| SCON2.6 | SM1 – UART2 mode bit 1 | |
| SCON2.5 | SM2 – Multiprocessor communication enable bit. In Mode 2 and 3, SM2 enables the multiprocessor communication. In Mode 2 or 3, when SM2=1, RI cannot be set in case that the received 9th bit is 0. In Mode 1, when SM2=1, RI can be set only if a valid stop bit is received. In Mode 0, SM2 establishes the baud rate: when SM2 = 0, the baud rate is clk/12; when SM2 = 1, the baud rate is clk. | |
| SCON2.4 | REN – Receive enable bit. When REN = 1, data receiving is enabled. | |
| SCON2.3 | TB8 – To define the 9th bit transmitted in Mode 2 and 3. | |
| SCON2.2 | RB8 – In Mode 2 and 3, RB8 indicates whether or not to store the received 9th bit. In Mode 1, RB8 indicates the received stop bit is stored. In Mode 0, RB8 is not used. | |
| SCON2.1 | TI – Transmit interrupt flag, indicating that data transmit data has been shifted out. In Mode 0, TI is set at the end of the 8th bit. In all other modes, TI is set when the stop bit is placed on the pin TXD2. TI must be cleared by the program. | |
| SCON2.0 | RI – Receive interrupt flag, indicating that a serial of data have been received. In Mode 0, RI is set after receiving the 8th data bit. In Mode 1, according to the state of SM2, RI is set after the last sample of the incoming stop bit. In Mode 2 and 3, RI is set at the | |

| | |
|---|---|
| | end of the last sample of the 9th bit. RI must be cleared by the program. |

**Table 11-8 UART2 Buffer Register (SBUF2, 0x1607)**

| 0x1607, R/W | UART2 Buffer Register, SBUF2 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- SBUF2 is physically two registers. One is written only and is used to hold data to be transmitted out of the MCU via the pin TXD2. The other is read only and is used to hold received data from external sources via the pin RXD2. Both mutually exclusive registers use the address 0x1607.

# 11.3. UART3

UART3 is the same as UART2 except that it cannot be used to transmit 38kHz carrier wave. There is a baud rate generator (compatible with Timer1) within UART3. As an additional peripheral, there is a dedicated control/status register that controls its baud rate, selects the clock source, on/off status, and overflow status of the baud rate generator, etc.

The registers associated with UART3 are listed in Table 11-9.

**Table 11-9 Registers Related to UART3**

| | |
|---|---|
| 0x1700, R/W | TCON3, UART3 Control / Status Register |
| 0x1701, R/W | TMOD3, UART3 Baud Rate Generator Mode Control Register |
| 0x1703, R/W | TH31, Higher Byte of Baud Rate Generator of UART3 |
| 0x1705, R/W | TL31, Lower Byte of Baud Rate Generator of UART3 |
| 0x1706, R/W | SCON3, UART3 Control Register |
| 0x1707, R/W | SBUF3, UART3 Buffer Register |

**Table 11-10 UART3 Control/ Status Register (TCON3, 0x1700)**

| 0x1700, R/W | UART3 Control/ Status Register, TCON3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | SMOD | - | T1M | - | TF1 | - | TR1 | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- SMOD: double UART3 baud rate enable bit. SMOD=1, UART3 baud rate doubles;
- T1M: to select the clock source for baud rate generator; 0, to use clk/12; 1, to use clk;
- TF1: baud rate generator overflow flag, no influence on the overflow interrupt. 1, overflow; 0, non-overflow.
- TR1: baud rate generator run control bit. 1, to run; 0, to stop.

# 11.4. UART4

UART4 is the same as UART2 except that it cannot be used to transmit 38kHz carrier wave. There is a baud rate generator (compatible with Timer1) within UART4. As an additional peripheral, there is a dedicated control/status register that controls its baud rate, selects the clock source, on/off status, and overflow status of the baud rate generator, etc.

The registers associated with UART4 are listed in Table 11-11.

**Table 11-11 Registers Related to UART4**

| | |
|---|---|
| 0x1800, R/W | TCON4, UART4 Control / Status Register |
| 0x1801, R/W | TMOD4, UART4 Baud Rate Generator Mode Control Register |
| 0x1803, R/W | TH41, Higher Byte of Baud Rate Generator of UART4 |
| 0x1805, R/W | TL41, Lower Byte of Baud Rate Generator of UART4 |
| 0x1806, R/W | SCON4, UART4 Control Register |
| 0x1807, R/W | SBUF4, UART4 Buffer Register |

**Table 11-12 UART4 Control/ Status Register (TCON4, 0x1800)**

| 0x1800, R/W | UART4 Control/ Status Register, TCON4 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | SMOD | - | T1M | - | TF1 | - | TR1 | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- SMOD: double UART4 baud rate enable bit. SMOD=1, UART4 baud rate doubles;
- T1M: to select the clock source for baud rate generator; 0, to use clk/12; 1, to use clk;
- TF1: baud rate generator overflow flag, no influence on the overflow interrupt. 1, overflow; 0, non-overflow.

- TR1: baud rate generator run control bit. 1, to run; 0, to stop.

# 11.5. UART5

UART5 is the same as UART2 except that it cannot be used to transmit 38kHz carrier wave. There is a baud rate generator (compatible with Timer1) within UART5. As an additional peripheral, there is a dedicated control/status register that controls its baud rate, selects the clock source, on/off status, and overflow status of the baud rate generator, etc.

The registers associated with UART5 are listed in Table 11-13.

**Table 11-13 Registers Related to UART5**

| | |
|---|---|
| 0x1900, R/W | TCON5, UART5 Control / Status Register |
| 0x1901, R/W | TMOD5, UART5 Baud Rate Generator Mode Control Register |
| 0x1903, R/W | TH51, Higher Byte of Baud Rate Generator of UART5 |
| 0x1905, R/W | TL51, Lower Byte of Baud Rate Generator of UART5 |
| 0x1906, R/W | SCON5, UART5 Control Register |
| 0x1907, R/W | SBUF5, UART5 Buffer Register |

**Table 11-14 UART5 Control/ Status Register (TCON5, 0x1900)**

| 0x1900, R/W | UART5 Control/ Status Register, TCON5 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | SMOD | - | T1M | - | TF1 | - | TR1 | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- SMOD: double UART5 baud rate enable bit. SMOD=1, UART5 baud rate doubles;
- T1M: to select the clock source for baud rate generator; 0, to use clk/12; 1, to use clk;
- TF1: baud rate generator overflow flag, no influence on the overflow interrupt. 1, overflow; 0, non-overflow.
- TR1: baud rate generator run control bit. 1, to run; 0, to stop.

# 11.6. UART Modes

Use the mode select bits, for example, SM1 and SM2 of the register SCON2 (0x1606), to configure the UART serial interfaces to work in different modes.

**Table 11-15 UART Modes**

| Mode | | Sync./Async. | Clock for Baud rate | Data Bits | Start/Stop Bit | The 9th Bit |
|---|---|---|---|---|---|---|
| 0 | 8-bit shift register | Sync. | UART0/1: clk/4 or clk/12<br>UART2~5: clk or clk/12 | 8 | None | None |
| 1 | 8-bit UART | Async. | Baud rate generator | 8 | 1 start, 1 stop | None |
| 2 | 9-bit UART | Async. | clk/32 or clk/64 | 9 | 1 start, 1 stop | 0, 1, parity |
| 3 | 9-bit UART | Async. | Baud rate generator | 9 | 1 start, 1 stop | 0, 1, parity |

UART0~UART5 works as the same as UART2 except that UART0 and UART1 use Timer1 and/or Timer2 as their baud rate generators, and UART0~UART1 and UART3~UART5 cannot transmit 38kHz carrier. So, take UART2 for an example to describe the work mode for UART serial interfaces.

## 11.6.1.Mode 0

In Mode 0, the UART can only receive data on the pin RXD2 (P1.4), and output shift clock on the pin TXD2 (P1.5). Data can be received as soon as the bit REN (SCON2.4) is set and the bit RI (SCON2.0) is cleared. The shift clock is activated and the UART2 shifts data in on each rising edge of the shift clock until eight bits have been received. The 8th bit was shifted in, and one machine cycle later, the bit RI is set and the reception stops until the bit RI is cleared by the program.

## 11.6.2.Mode 1

Mode 1 provides standard asynchronous and full-duplex communication. In this mode, a data frame includes ten bits: one start bit, eight data bits, and one stop bit. As to receiving operation, the stop bit is stored in the bit RB8 (SCON2.2). When data bits are received and transmitted, start with the LSB.

In Mode 1, UART2 begins to transmit data after the program writing data into the register SBUF2 (0x1607). UART2 transmits data on the pin TXD2 (P1.5) in the following order: start bit, eight data bits (LSB first), stop bit. The bit TI (SCON2.1) is set after the stop bit has been transmitted.

In Mode 1, UART2 starts to receive data at the falling edge of a start bit received on the pin RXD2

(P1.4), when REN=1 (SCON2.4). To achieve it, every bit on the pin RXD2 (P1.4) should be sampled sixteen times at any baud rate. When a falling edge of a start bit is detected, the timer used to generate the receive clock is reset to synchronize with the received bits. To reject noise, the serial port detects the values of the three consecutive samples in the middle of every bit. Only when more than two values are the same, the received data bit is verified to be valid. This is especially true for the start bit. If the falling edge on the pin RXD2 (P1.4) is not verified by a majority decision of three consecutive samples, then the serial port stops receiving data and waits for another falling edge on RXD2 (P1.4).

When RI = 0 (SCON2.0), SM2 = 1 (SCON2.5), and the stop bit is 1 (if SM2 = 0, the state of stop bit does not matter), the serial port will write the received byte to the register SBUF2 (0x1607), load the stop bit into RB8 (SCON2.2), and set the bit RI (SCON2.0). Otherwise, the received data lose, they cannot load data into the register SBUF2 and the bit RB8, and the bit RI cannot be set.

In Mode 1, the baud rate is a function of baud rate generator overflow frequency. UART0 uses either Timer1 or Timer2 as the baud rate generator, UART1 uses only Timer1 as the baud rate generator, and UART2~UART5 use dedicated baud rate generators, which are compatible with Timer 1. When the baud rate generator overflows, a clock is sent to the baud rate circuit. The clock is then divided by 16 to generate the baud rate.

Take Timer1 and Timer2 as examples to describe how to calculate the baud rate.

1. To use Timer1 as the baud rate generator

When Timer1 is used as the baud rate generator, the baud rate can be given by the following formula.

$$BaudRate = \frac{2^{SMODx}}{32} \times Overflow \qquad \text{Equation 11-1}$$

where, Overflow is the overflow frequency of Timer1; SMODx is the value of the bits SMOD0/1 which enables or disables the Timer1 overflow frequency being divided by 2, which determines whether or not double the baud rate, as shown in Figure 10-9.

When Timer1 is used as the baud rate generator, the 8-bit auto-reload mode is commonly used. The reload value is stored in the register TH1 SFR (0x8D), which makes the complete formula for baud rate (clk/12 is selected, CKCON.4=0):

$$BaudRate = \frac{2^{SMODx}}{32} \times \frac{Clk}{12 \times (256 - TH1)} \qquad \text{Equation 11-2}$$

where, clk is the system clock (MCU frequency), and TH1 is the value of the register TH1 SFR (0x8D).

The bit T1M (CKCON.4) determines the clock source of Timer1. When T1M=1, clk is selected.

$$BaudRate = \frac{2^{SMODx}}{32} \times \frac{Clk}{(256 - TH1)}$$ Equation 11-3

When the baud rate is known, users can obtain the value of TH1 via Formula 11-4

$$TH1 = 256 - \frac{2^{SMODx} \times clk}{32 \times BaudRate}$$ Equation 11-4

When T1M=0, and the baud rate is known, users can obtain the value of TH1 via Formula 11-5:

$$TH1 = 256 - \frac{2^{SMODx} \times clk}{384 \times BaudRate}$$ Equation 11-5

Likewise, users can calculate the baud rate of UART2~UART5.

Table 11-16 lists the relationship between baudrate, SMOD configuration and the value of TH1 at different MCU frequencies.

**Table 11-16 Baudrate and TH1 at Different MCU Frequencies**

| MCU Frequency | | SMODx=0 | | SMODx=1 | |
|---|---|---|---|---|---|
| | Baudrate | 1/12 clk | 1 clk | 1/12 clk | 1 clk |
| 13107200 | 300 | 142.2222 | -1109.33 | 28.44444 | -2474.67 |
| | 600 | 199.1111 | -426.667 | 142.2222 | -1109.33 |
| | 1200 | 227.5556 | -85.3333 | 199.1111 | -426.667 |
| | 2400 | 241.7778 | 85.33333 | 227.5556 | -85.3333 |
| | 4800 | 248.8889 | 170.6667 | 241.7778 | 85.33333 |
| | 9600 | 252.4444 | 213.3333 | 248.8889 | 170.6667 |
| | 19200 | 254.2222 | 234.6667 | 252.4444 | 213.3333 |
| ` | | | | | |
| 6553600 | 300 | 199.1111 | -426.667 | 142.2222 | -1109.33 |
| | 600 | 227.5556 | -85.3333 | 199.1111 | -426.667 |
| | 1200 | 241.7778 | 85.33333 | 227.5556 | -85.3333 |
| | 2400 | 248.8889 | 170.6667 | 241.7778 | 85.33333 |
| | 4800 | 252.4444 | 213.3333 | 248.8889 | 170.6667 |
| | 9600 | 254.2222 | 234.6667 | 252.4444 | 213.3333 |
| | 19200 | 255.1111 | 245.3333 | 254.2222 | 234.6667 |

| | | | | | |
|---|---|---|---|---|---|
| ` | | | | | |
| 3276800 | 300 | 227.5556 | -85.3333 | 199.1111 | -426.667 |
| | 600 | 241.7778 | 85.33333 | 227.5556 | -85.3333 |
| | 1200 | 248.8889 | 170.6667 | 241.7778 | 85.33333 |
| | 2400 | 252.4444 | 213.3333 | 248.8889 | 170.6667 |
| | 4800 | 254.2222 | 234.6667 | 252.4444 | 213.3333 |
| | 9600 | 255.1111 | 245.3333 | 254.2222 | 234.6667 |
| | 19200 | 255.5556 | 250.6667 | 255.1111 | 245.3333 |
| ` | | | | | |
| 1638400 | 300 | 241.7778 | 85.33333 | 227.5556 | -85.3333 |
| | 600 | 248.8889 | 170.6667 | 241.7778 | 85.33333 |
| | 1200 | 252.4444 | 213.3333 | 248.8889 | 170.6667 |
| | 2400 | 254.2222 | 234.6667 | 252.4444 | 213.3333 |
| | 4800 | 255.1111 | 245.3333 | 254.2222 | 234.6667 |
| | 9600 | 255.5556 | 250.6667 | 255.1111 | 245.3333 |
| | 19200 | 255.7778 | 253.3333 | 255.5556 | 250.6667 |

2. To use Timer2 as the baud rate generator

When Timer2 is used as the baud rate generator for UART0, the baud rate can be given by the following formula.

$$BaudRate = \frac{Timer2Overflow}{16} \qquad \text{Equation 11-6}$$

where, Timer2Overflow is the overflow frequency of Timer2.

When either TCLK or RCLK is set, Timer2 is forced into baud rate generation mode. In this mode, Timer2 runs at a divide by 2 in the 16-bit auto-reload mode. Thus the baud rate can be given by Formula 11-7.

$$BaudRate = \frac{1}{16} \times \frac{Clk}{2 \times (65536 - RCAP2H, RCAP2L)} \qquad \text{Equation 11-7}$$

where, clk is the system clock (MCU frequency), and (RCAP2H, RCAP2L) is the preset values of the registers RCAP2H SFR and RCAP2L SFR which will be automatically reloaded into the registers TH2 SFR and TL2 SFR.

When the desired baud rate is known, users can use the following formula to get the reload value.

$$(RCAP2H, RCAP2L) = 65536 - \frac{clk}{32 \times BaudRate} \qquad \text{Equation 11-8}$$

## 11.6.3.Mode 2

Mode 2 provides asynchronous and full-duplex communication. In this mode, the data frame includes eleven bits: 1 start bit, eight data bits, one programmable 9th bit, and one stop bit.

When data bits are received and transmitted, start with LSB. As to the transmitting operation, the 9th bit is determined by the value of the bit TB8 (SCON2.3). If the 9th bit is used as a parity bit, the value of the P bit (Bit 0 of PSW SFR) should be moved to TB8.

In Mode 2, receiving data begins at the falling edge of a start bit received on the pin RXD2 (P1.4), when the bit REN=1 (SCON2.4). To achieve it, every bit on the pin RXD2 (P1.4) should be sampled sixteen times at any baudrate. When a falling edge of a start bit is detected, the timer used to generate the receive clock is reset to synchronize with the received bits. To reject noise, the serial port detects the values of the three consecutive samples in the middle of every bit. Only when more than two values are the same, the received data bit is verified to be valid. This is especially true for the start bit. If the falling edge on the pin RXD2 (P1.4) is not verified by a majority decision of three consecutive samples, then the serial port stops receiving data and waits for another falling edge on RXD2 (P1.4).

When RI = 0 (SCON2.0), SM2 = 1 (SCON2.5), and the stop bit is 1 (if SM2 = 0, the state of stop bit does not matter), the serial port will write the received byte to the register SBUF2 (0x1607), load the stop bit into RB8 (SCON2.2), and set the bit RI (SCON2.0). Otherwise, the received data lose, they cannot load data into the register SBUF2 and the bit RB8, and the bit RI cannot be set.

In Mode 2, the baudrate is either clk/32 or clk/64, determined by the bit SMOD (Bit 7) in the register TCON2. It can be calculated as follows.

$$BaudRate = \frac{2^{SMODx} \times clk}{64} \qquad \text{Equation 11-9}$$

In Mode 2, UART2 starts transmitting data after the software writing data into the register SBUF2 (0x1607). UART2 transmits data on the pin TXD2 (P1.5) in the following order: the start bit, eight data bits (LSB first), the 9th bit, then, the stop bit. The bit TI (SCON2.1) is set when the stop bit has been transmitted.

## 11.6.4.Mode 3

Mode 3 provides asynchronous and full-duplex communication. In this mode, the data frame includes eleven bits: 1 start bit, eight data bits, one programmable 9th bit, and one stop bit. When

data bits are received and transmitted, start with LSB.

In Mode 3, the data is transmitted and received in the same way that in Mode 2. In Mode 3, the baud rate generation is identical to that in Mode 1. That is, Mode 3 is a combination of the protocol in Mode 2 and the baud rate in Mode 1.

### 11.6.5.Multiprocessor Communications

The multiprocessor communication is enabled in Mode 2 and 3 when the bit SM2 (SCON2.5) is set. In the multiprocessor communication mode, the received 9$^{th}$ bit is stored in the bit RB8 (SCON2.2), and, after the stop bit has been received, UART2 receive interrupt is activated only if RB8 = 1.

The multiprocessor communication is used to send a block of data from a master to one slave. The master first transmits an address byte that identifies the target slave. When transmitting an address byte, the master sets the 9$^{th}$ bit to 1; when transmitting data bytes, the 9$^{th}$ bit is set to 0.

When SM2 = 1 (SCON2.5), no slave can generate an interrupt when a data byte has been received. However, all slaves can generate interrupts when an address byte is received. Every slave can examine the received address byte to determine whether it is the slave being addressed. Address decoding must be done by the program during the interrupt service routine. The addressed slave clears the bit SM2 and prepares to receive the data bytes. The slaves that are not being addressed leave the bit SM2 set and ignore the incoming data bytes.

## 11.7. IR Modulation/Demodulation

In V9003/V9103, UART2 can be used for IR communication.

## 11.7.1.Instruction

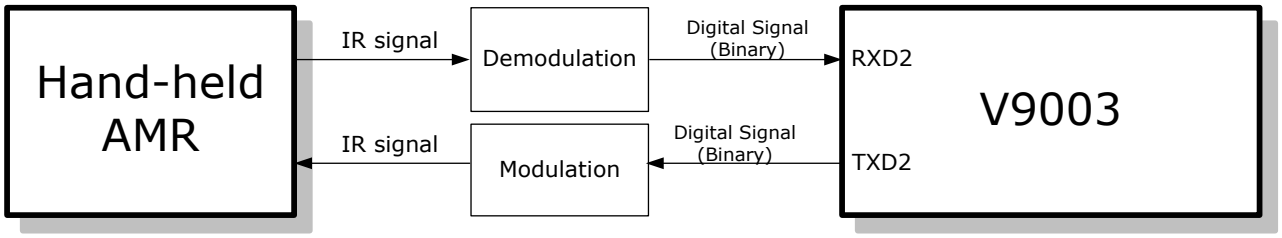



**Figure 11-1 IR Communication Interfaces**

On the pin TXD2, the digital signal (in binary) is modulated into an IR signal of a certain frequency, and is transmitted by the IR transmitter in the form of light pulse. The digital signal "1" means no IR pulse is transmitted; the digital signal "0" means an IR pulse is transmitted, as shown in Figure 11-2. In V9003/V9103, in IR communication, a 38.1kHz carrier wave is obtained via dividing 819.2kHz by 22, of which the duty cycle can be configured in the register IRDUTY (0x1B46), over the range of 0~100%.

On the pin RXD2, the received IR signal is demodulated into the digital signal. No IR signal means the digital signal "1" is received; an IR signal means the digital signal "0" is received.

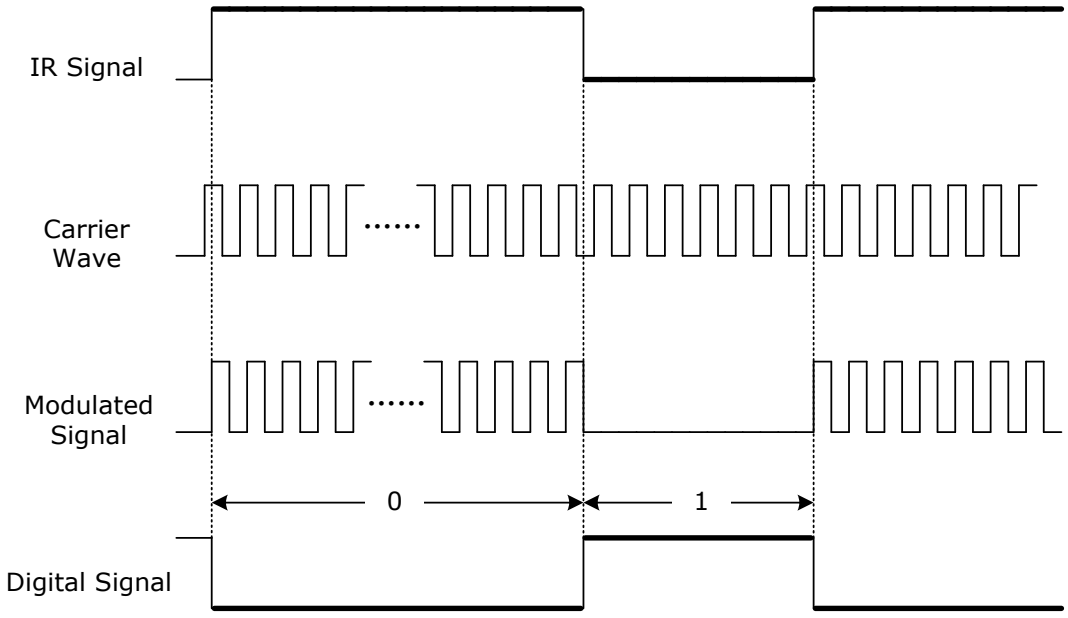



**Figure 11-2 Signal Conversion in IR Communication**

In V9003/V9103, the IR signal is sampled 819200 times per second (the sampling frequency = 819.2kHz, 1/8 of MCU frequency, $f_{MCU}$=6.5536MHz), the baud rate is no more than 1.2kbps, the pulse of the modulated signal is over the range of 36~40kHz, an IR pulse is sampled 20~23 times, and a bit "1" includes 30~34 modulated signal pulses.

As shown in Figure 13-3, when a 38.1kHz carrier wave is used, and baudrate is 1.2kbps, a bit "0" represents 32 modulated signal pulses.

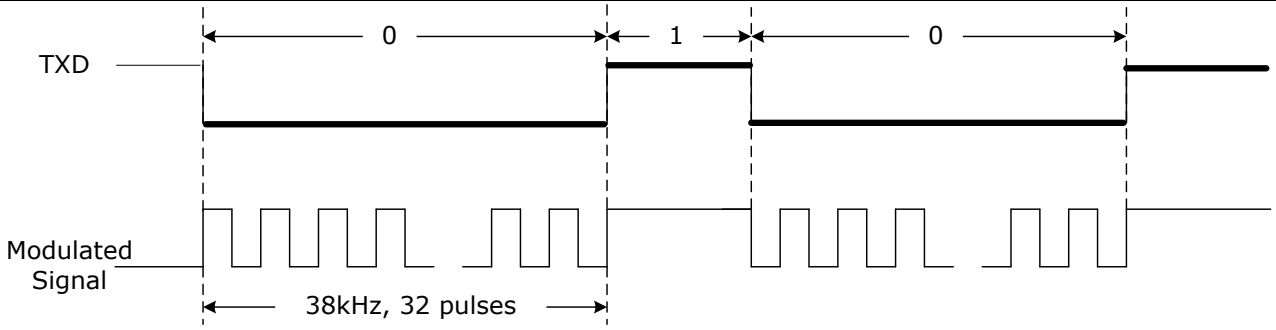



**Figure 11-3 Modulating a Signal with 38.1kHz Carrier Wave**

## 11.7.2.IR Tranmission and Reception

In V9003/V9103, IR reception and transmission modules work as follows:

1. To detect whether there is an IR signal in the time window via three parameters (as shown in Figure 11-4):

   1.1 The time window width, namely, how many times an IR signal is sampled, which should be over the range of 20~23, configured by the register IRSMP (0x1B45);

   1.2 The minimum number of the low level of the modulated signal pulse (0);

   1.3 The minimum number of the high level of the modulated signal pulse (1).

When the minimum number of 1 and 0 of the modulated signal pulse is met, an IR signal pulse is received or transmitted in the time window.

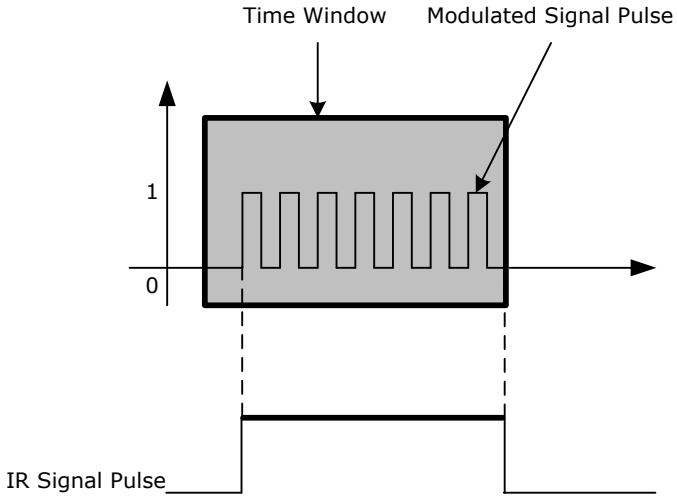



**Figure 11-4 Detecting an IR Signal**

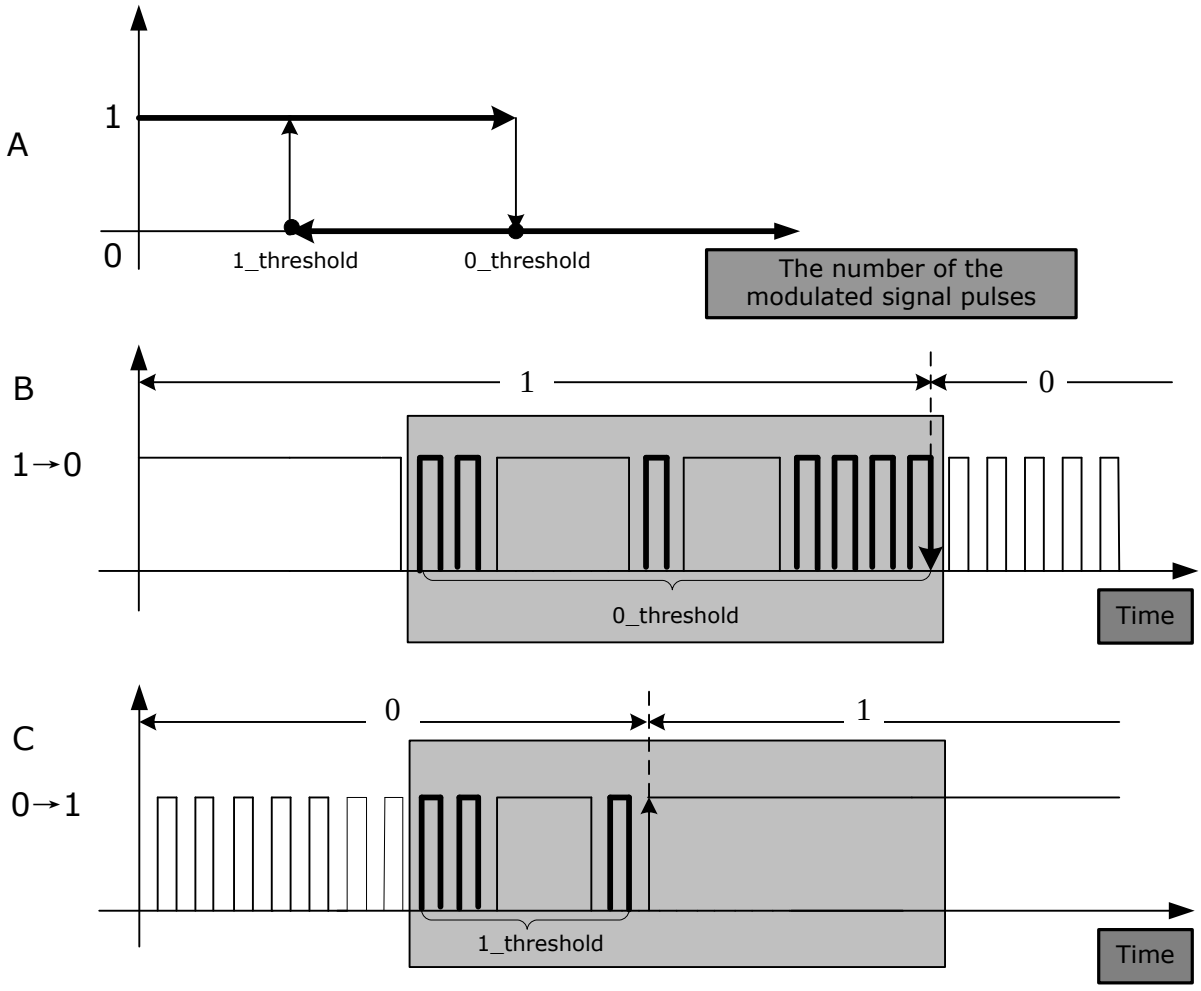
2. To detect the demodulating data based on the number of the modulated signal pulses in a 20-bit time window. In V9003/V9103, the demodulation is monostable, and 2 configurable

parameters, 1_threshold and 0_threshold, can be used to detect the demodulating data. For example, as shown in Figure 11-5, when 1_threshold=3, and 0_threshold=7,

2.1 as shown in B of Figure 11-5, when more than 7 the modulated signal pulses (namely, 0_threshold) moved into the time window when the demodulating data is 1, the demodulating data will be switched over to 0;

2.2 as shown in C of Figure 11-5, when less than 3 the modulated signal pulses (namely, 1_threshold) moved into the time window when the demodulating data is 0, the demodulating data will be switched over to 1.

The parameters, 1_threshold and 0_threshold, can be configured via the registers IR1TH (0x1B43) and IR0TH (0x1B44), see "IR control registers" for details.




1, no IR signal
0, IR signal
The gray frame representing the time window

**Figure 11-5 Demodulation of IR Signals**

## 11.7.3.IR Control Registers

Table 11-17 to Table 11-23 list all related registers to IR modulation/demodulation control.

**Table 11-17 IR Control Register (IRCON, 0x1B40)**

| 0x1B40, R/W | IR Control Register, IRCON | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| - | - | - | - | - | - | IRS | IRT | IRR |
| Default Value | - | - | - | - | - | 0 | 0 | 0 |

- Write data into the register SBUF2 (0x1607) to start transmitting serial data via UART2. When the data has been transmitted, the transmitting interrupt flag (TI, SCON2.1) is set. Set the bit REN (SCON2.4) to start data receiving via UART2. When the data has been received, the receiving interrupt flag (RI, SCON2.0) is set.
- IRR: 1, to receive demodulated data; 0, to receive unmodulated data.
- IRT: 1, to transmit modulated data; 0, to transmit unmodulated data.
- IRS: 1, to demodulate IR data from analog input port; 0, to demodulate IR data from the pin RXD2 (P1.4). In V9003/V9103, IRS must be set to 0.

**Table 11-18 Minimum High Levels on Modulated Signal Pulse Control Register (IRONE, 0x1B41)**

| 0x1B41, R/W | Minimum High Levels on Modulated Signal Pulse Control Register, IRONE | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| - | - | - | - | | | | | |
| Default Value | - | - | - | 0 | 0 | 0 | 0 | 0 |

The due minimum number of the high level (1) when there are modulated signal pulses in the time window. 7 is recommended.

**Table 11-19 Minimum Low Levels on Modulated Signal Pulse Control Register (IRZERO, 0x1B42)**

| 0x1B42, R/W | Minimum Low Levels on Modulated Signal Pulse Control Register, IRZERO | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| - | - | - | - | | | | | |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Default Value | - | - | - | 0 | 0 | 0 | 0 | 0 |

The due minimum number of the low level (0) when there are modulated signal pulses in the time window. 7 is recommended.

**Table 11-20 IR 1_threshold Register (IR1TH, 0x1B43)**

| 0x1B43, R/W | IR 1_threshold Register, IR1TH | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| - | - | - | - | | | | | |
| Default Value | - | - | - | 0 | 0 | 0 | 0 | 0 |

The maximum number of the modulated signal pulses in the time window to switch the demodulating data from 0 to 1. 10 is recommended.

**Table 11-21 IR 0_threshold Register (IR0TH, 0x1B44)**

| 0x1B44, R/W | IR 0_threshold Register, IR0TH | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| - | - | - | - | | | | | |
| Default Value | - | - | - | 0 | 0 | 0 | 0 | 0 |

The minimum number of the modulated signal pulses in the time window to switch the demodulating data from 1 to 0. 10 is recommended.

**Table 11-22 Time Window Width Control Register (IRSMP, 0x1B45)**

| 0x1B45, R/W | Time Window Width Control Register, IRSMP | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| - | - | - | - | | | | | |
| Default Value | - | - | - | 0 | 0 | 0 | 0 | 0 |

The sampling times on an IR signal pulse. 22 is recommended.

**Table 11-23 Carrier Wave Duty Cycle Control Register (IRDUTY, 0x1B46)**

| 0x1B46, R/W | Carrier Wave Duty Cycle Control Register, IRDUTY | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | | | | | | | | |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Configuration of the duty cycle of 38kHz carrier wave. 11 (50%) is recommended.

# 12. LCD

## 12.1. LCD Timing and Driving Circuits

LCD timing and driving circuits use the independent 32.768 kHz clock. POR/BOR and RST Pin Reset can reset the LCD module. After reset, the driving circuit of LCD is turned off, the scanning frequency is 512Hz, the impedance of each resistor in the driving ladder is 100KΩ, and all the display buffers are cleared. The LCD module can be turned off by the register LCDG (0x1210) and the register LCDCtrl (0x1F14) to lower power consumption. Control bits of the LCD driving voltage (3V, 3.3V, or 3.5V) are located in the registers LDOCFG, V3P and V3P5.

LCD can generate control signals to drive up to 40 SEGs, 32 of which share PAD with P5~P8 ports (see also GPIO). Every SEG signal output can be enabled or disabled independently.

All registers in the LCD module is 8-bit wide, and the addresses of them begins with 0x1F00. 0x1F00~0x1F13 are the addressed for the display buffer registers, and 0x1F14 is the address of the display control register. 0x1F15~0x1F19 is the addresses of the SEG control registers.

**Table 12-1 Display Control Register (LCDCtrl, 0x1F14)**

| 0x1F14, R/W | Display Control Register, LCDCtrl | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | Bit4 | bit3 | bit2 | bit1 | bit0 |
| | ON/OFF | | VOLTAGE | TYPE | DRV1 | DRV0 | FRQ1 | FRQ0 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- FRQ1, FRQ0: to configure the scanning frequency. 11, 512Hz; 10, 256Hz; 01, 128Hz; 00, 64Hz.
- DRV1, DRV0: to configure the impedance of each resistor in the driving ladder. 00, 100kΩ; 01, 70kΩ; 10, 40kΩ; 11, 10kΩ.
- TYPE: 0, to use a resistor ladder as the driving circuit; 1, to use a charge pump as the driving circuit.
- ON/OFF: to turn on/turn off the analog LCD driving circuits. 1, to turn on the driving circuit, the analog driving circuits outputting signals to COM and SEG; otherwise, outputting high impedance.
- VOLTAGE: must be 0. Otherwise, higher power consumption occurs.

**Table 12-2 SEG Control Register 0 (SegCtrl0, 0x1F15)**

| 0x1F15, R/W | Seg Control Register 0, SegCtrl0 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | Bit4 | bit3 | bit2 | bit1 | bit0 |
| | SEGON7 | SEGON6 | SEGON5 | SEGON4 | SEGON3 | SEGON2 | SEGON1 | SEGON0 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 12-3 SEG Control Register 1 (SegCtrl1, 0x1F16)**

| 0x1F16, R/W | Seg Control Register 1, SegCtrl1 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | Bit4 | bit3 | bit2 | bit1 | bit0 |
| | SEGON 15 | SEGON 14 | SEGON 13 | SEGON 12 | SEGON 11 | SEGON 10 | SEGON9 | SEGON8 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 12-4 SEG Control Register 2 (SegCtrl2, 0x1F17)**

| 0x1F17, R/W | Seg Control Register 2, SegCtrl2 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | SEGON 23 | SEGON 22 | SEGON 21 | SEGON 20 | SEGON 19 | SEGON 18 | SEGON 17 | SEGON 16 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 12-5 SEG Control Register 3 (SegCtrl3, 0x1F18)**

| 0x1F18, R/W | SEG Control Register 3, SegCtrl3 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | SEGON 31 | SEGON 30 | SEGON 29 | SEGON 28 | SEGON 27 | SEGON 26 | SEGON 25 | SEGON 24 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 12-6 SEG Control Register 4 (SegCtrl4, 0x1F19)**

| 0x1F19, R/W | Seg Control Register 4, SegCtrl4 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | SEGON 39 | SEGON 38 | SEGON 37 | SEGON 36 | SEGON 35 | SEGON 34 | SEGON 33 | SEGON 32 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- SEGONx: 0, to disable SEG output; 1, to enable SEG output. P5~P8 ports can be multiplex with SEG output, when the SEG output is disabled, they can be used as general purpose IOs.

Seg(n) and Seg(n+1) are mapped to the lower 4 bits and higher 4 bits respectively in a display buffer register. The LCD display is updated simultaneously the display buffer registers are updated.

**Table 12-7 Display Buffer Register**

| Display Buffer Register | Register | SEG | D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Com3 | Com2 | Com1 | Com0 | Com3 | Com2 | Com1 | Com0 |
| 0x1F00 | LCDM00 | S01 S00 | Seg01 | | | | Seg00 | | | |
| 0x1F01 | LCDM01 | S03 S02 | Seg03 | | | | Seg02 | | | |
| 0x1F02 | LCDM02 | S05 S04 | Seg05 | | | | Seg04 | | | |
| 0x1F03 | LCDM03 | S07 S06 | Seg07 | | | | Seg06 | | | |
| 0x1F04 | LCDM04 | S09 S08 | Seg09 | | | | Seg08 | | | |
| 0x1F05 | LCDM05 | S11 S10 | Seg11 | | | | Seg10 | | | |
| 0x1F06 | LCDM06 | S13 S12 | Seg13 | | | | Seg12 | | | |
| 0x1F07 | LCDM07 | S15 S14 | Seg15 | | | | Seg14 | | | |
| 0x1F08 | LCDM08 | S17 S16 | Seg17 | | | | Seg16 | | | |
| 0x1F09 | LCDM09 | S19 S18 | Seg19 | | | | Seg18 | | | |
| 0x1F0a | LCDM10 | S21 S20 | Seg21 | | | | Seg20 | | | |
| 0x1F0b | LCDM11 | S23 S22 | Seg23 | | | | Seg22 | | | |
| 0x1F0c | LCDM12 | S25 S24 | Seg25 | | | | Seg24 | | | |
| 0x1F0d | LCDM13 | S27 S26 | Seg27 | | | | Seg26 | | | |

| | | | | |
|---|---|---|---|---|
| 0x1F0e | LCDM14 | S29 S28 | Seg29 | Seg28 |
| 0x1F0f | LCDM15 | S31 S30 | Seg31 | Seg30 |
| 0x1F10 | LCDM16 | S33 S32 | Seg33 | Seg32 |
| 0x1F11 | LCDM17 | S35 S34 | Seg35 | Seg34 |
| 0x1F12 | LCDM18 | S37 S36 | Seg37 | Seg36 |
| 0x1F13 | LCDM19 | S39 S38 | Seg39 | Seg38 |

# 12.2. Configuring LCD Driving Voltage

In V9003/V9103,3.0V, 3.3V and 3.5V can be used as driving voltage for LCD, which can be selected via configuring the bits V3P and V3P5 in the register LDOCFG (0x1B14). After POR/BOR, RST Pin Reset and IO/RTC Wake-up Reset, the default value of the driving voltage is 3.3V.

**Table 12-8 Related Bits for LCD Driving Voltage**

| V3P | V3P5 | LCD Driving Voltage |
|---|---|---|
| 0 | 1 | 3.5V |
| 1 | 0 | 3V |
| 0 | 0 | 3.3V |

# 13. GPIO

There are 8 general-purpose IO ports (GPIO) in V9003/V9103, P0, P1, P2, P3, P5, P6, P7 and P8. All of POR/BOR, RST Pin Reset, IO/RTC Wake-up Reset, WDT Reset and Debug Reset can reset GPIO into a default state: output disabled and input masked. Both input and output current on these IO ports should be ≥2mA. In Sleep or Deep Sleep, GPIO keeps its state before entering Sleep or Deep Sleep.

Every GPIO port has its output/input enable control registers. When MODE0 =0, P0 ports are used as JTAG interfaces regardless of the P0 output/input enable control registers. Special functions of IOs are selected by the IO function select registers. No matter with special function or not, the output and input on IO ports are controlled by the output enable register and the input enable register respectively.

P0 ports are multiplex with JTAG interfaces. P1, P2 and P3 ports are multiplex with special function IOs. P5~P8 ports are multiplex with SEG output (See "Pin Description" for functions of each IO pin).

The Pad type of all GPIOs is PDB04DGZ_APR. They can be set to digital input, digital output, digital input and output, and analog output. Table 13-1 shows the port description of PDB04DGZ_APR.

**Table 13-1 Port Description of PDB04DGZ_APR**

| Port | Description |
|---|---|
| EN | Input enable. 1, to enable input; 0, to disable input (input) |
| C | Input data (output) |
| OEN | Output enable. 1, to disable output; 0, to enable output (input) |
| I | Output data (input) |
| AI | Analog output data, connecting to the PAD directly (inout). |
| PAD | Pad, connecting to the chip (inout). |

Table 13-2 lists the true values of PDB04DGZ_ARP. The chip will work abnormally or even be burned up when it is configured as those shown in bold.

**Table 13-2 True Values of PDB04DGZ_APR**

| EN | OEN | AI | PAD | C | Description |
|---|---|---|---|---|---|
| 0 | 0 | Z | I | 1 | Digital output |
| **0** | **0** | **Analog Signal** | **X** | **1** | **Digital output interfering with analog output mutually.** |
| 0 | 1 | - | AI | 1 | Analog output |
| 1 | 0 | Z | I | I | Digital input and output |
| **1** | **0** | **Analog Signal** | **X** | **X** | **Digital output interfering with analog output mutually.** |
| **1** | **1** | **Analog Signal** | **AI** | **X** | **Analog signal is read on digital input.** |
| 1 | 1 | Z | PAD | PAD | Digital input |

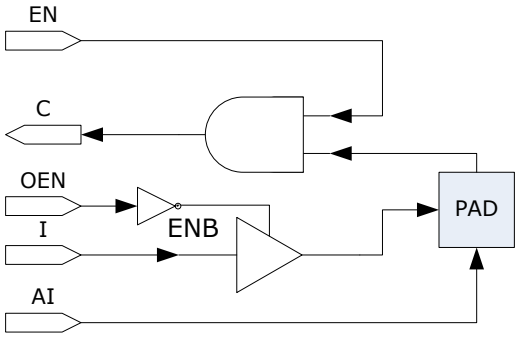



**Figure 13-1 Architecture of PDB04DGZ_APR**

When PDB04DGZ_APR is used as a digital IO port, AI must be floating or highly resistive. When PDB04DGZ_APR is used as an analog IO port, digital input and output must be disabled.

To decrease the number of pads, P0~P3, and P5~P8, not only can be used for digital input and output, but also be used for special functions. Use the register PxFS (x=1, 2, and 3) to select the working mode (except P0). In different working modes, connect the corresponding pad of GPIOs to the internal modules in the chip, and these pads can be configured for input or output. If these pads are not configured for the corresponding modes, 0 or 1 is input to these modules to protect the internal circuit from abnormal working because of some connection of the pads. In any mode, the registers PxIE and PxOEN (x=0~3, and 5~8) decide the input and output enable of the pads. Table 13-3 lists the port configuration and related registers of GPIO.

**Table 13-3 Port Configuration and Related Registers of GPIOs**

| | Port | Register | Description |
|---|---|---|---|
| P0~P3<br>P5~P8 | EN | PxIE | 1, to enable input;<br>0, input is read as 1, constantly. |
| | OEN | PxOEN | 1, to output high impedance; 0, to enable output. |
| | AI | From the analog circuit, or floating. | |
| | C | PxIN | All read-in signals are synchronized by two triggers. All clocks are synchronized with the MCU frequency. |
| | I | PxOUT | The output signal is stored via a trigger. |
| | Description | x = 0, 1, 2, 3, 5, 6, 7 and 8 | |

# 13.1. P0

The four P0 ports are multiplex with debug ports. To ensure safety, the working mode of P0 should be determined by the pin MODE0.

**Table 13-4 Special Function Configuration of P0**

| Pin | MODE0 | |
|---|---|---|
| | 0 | 1 |
| P0.0/TMS | TMS(I) | GPIO |
| P0.1/TDI | TDI(I) | |
| P0.2/TDO | TDO(O) | |
| P0.3/TCK | TCK(I) | |

**Table 13-5 Special Function Configuration of P0 in Mode 0**

| Pin | MODE0 | IE | OEN | I | C | AI |
|---|---|---|---|---|---|---|
| P0.0/TMS | 0 | 1 | 1 | P0OUT[0] | TMS | Floating |
| P0.1/TDI | | 1 | 1 | P0OUT[1] | TDI | |
| P0.2/TDO | | 0 | 0 | TDO | | |
| P0.3/TCK | | 1 | 1 | P0OUT[3] | TCK | |

**Table 13-6 P0 Input Enable Register (P0IE, 0x1A00)**

| 0x1A00, R/W | P0 Input Enable Register, P0IE | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | P03INEN | P02INEN | P01INEN | P00INEN |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 1, to enable input; 0, to disable input
- When MODE0=0, bit0, bit1 and bit3 are set to 1, and bit2 is set to 0.

**Table 13-7 P0 Output Enable Register (P0OEN, 0x1A01)**

| 0x1A01, R/W | P0 Output Enable Register, P0OEN | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | Bit1 | bit0 |
| | - | - | - | - | P03OEN | P02OEN | P01OEN | P00OEN |
| Default Value | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

- 1, to disable output; 0, to enable output
- When MODE0=0, bit0, bit1 and bit3 are set to 1, and bit2 is set to 0.

**Table 13-8 P0 Input Data Register (P0ID, 0x1A02)**

| 0x1A02, R/W | P0 Input Data Register, P0ID | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | 0 | 0 | 0 | 0 | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 13-9 P0 Output Data Register (P0OD, 0x1A03)**

| 0x1A03, R/W | P0 Output Data Register, P0OD | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |

## 13.2. P1

The eight P1 ports are multiplex with special function IOs, of which, P1.0 and P1.2 can be used to

wake up the system from Sleep when they were configured to input. No matter with special function or not, the output and input on IO ports are controlled by output enable register and input enable register respectively. Users can use the special function select register to configure the special functions of each IO port.

**Table 13-10 P1 Input Enable Register (P1IE, 0x1A10)**

| 0x1A10, R/W | P1 Input Enable Register, P1IE | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 1, to enable input; 0, to disable input

**Table 13-11 P1 Output Enable Register (P1OEN, 0x1A11)**

| 0x1A11, R/W | P1 Output Enable Register, P1OEN | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

- 1, to disable output; 0, to enable output

**Table 13-12 P1 Input Data Register (P1ID, 0x1A12)**

| 0x1A12, R/W | P1 Input Data Register, P1ID | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Table 13-13 P1 Output Data Register (P1OD, 0x1A13)**

| 0x1A13, R/W | P1 Output Data Register, P1OD | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Table 13-14 P1 Special Function Select Register 1 (P1FSEL, 0x1A15)**

| 0x1A15, R/W | P1 Special Function Select Register 1, P1FSEL |
|---|---|

| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
|---|---|---|---|---|---|---|---|---|
| | P17FSEL | P16FSEL | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- Bit5~Bit0 must be 0;
- P16FSEL: 0, RXD3, for UART3; 1, PLL clock output
- P17FSEL：0, TXD3, for UART3; 1, OSC clock output.

**Table 13-15 P1 Special Function Select Register 2 (P1FS, 0x1A14)**

| 0x1A14, R/W | P1 Special Function Select Register 2, P1FS | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | P17FNC | P16FNC | P15FNC | P14FNC | P13FNC | P12FNC | P11FNC | P10FNC |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 0, GPIO
- 1, Special function. P17FNC~P10FNC determines the special functions of the ports P1.7~P1.0, see Table 13-16 for details.

**Table 13-16 Function Configuration of P1**

| P1 Ports | P1FSEL | | P1FS | | |
|---|---|---|---|---|---|
| | Bit | | Bit | 1 | 0 |
| P1.0 | - | Must be 0 | P10FNC | RXD0 (I), UART0 RX | GPIO |
| P1.1 | - | Must be 0 | P11FNC | TXD0 (O), UART0 TX | |
| P1.2 | - | Must be 0 | P12FNC | RXD1 (I), UART1 RX | |
| P1.3 | - | Must be 0 | P13FNC | TXD1 (O), UART1 TX | |
| P1.4 | - | Must be 0 | P14FNC | RXD2 (I), UART2 RX | |
| P1.5 | - | Must be 0 | P15FNC | TXD2 (O), UART2 TX | |
| P1.6 | P16FNSEL | 1, PLL clock output; 0, RXD3, for UART3 | P16FNC | RXD3 (I), UART3 RX<br>PLL clock output | |
| P1.7 | P17FNSEL | 1, OSC clock output; 0, TXD3, for UART3 | P17FNC | TXD3 (O), UART3 TX<br>OSC clock output | |

# 13.3. P2

The 8 P2 ports are multiplex with special function IOs, of which, P2.6 can be used to wake up the system from Sleep or Deep Sleep when it was configured to input. No matter with special function or not, the output and input on IO port are controlled by output enable register and input enable register respectively. Users can use the special function select register to configure the special functions of each IO port.

**Table 13-17 P2 Input Enable Register (P2IE, 0x1A20)**

| 0x1A20, R/W | P2 Input Enable Register, P2IE | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 1, to enable input; 0, to disable input

**Table 13-18 P2 Output Enable Register (P2OEN, 0x1A21)**

| 0x1A21, R/W | P2 Output Enable Register, P2OEN | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

- 1, to disable output; 0, to enable output

**Table 13-19 P2 Input Data Register (P2ID, 0x1A22)**

| 0x1A22, R/W | P2 Input Data Register, P2ID | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Table 13-20 P2 Output Data Register (P2OD, 0x1A23)**

| 0x1A23, R/W | P2 Output Data Register, P2OD | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Table 13-21 P2 Special Function Select Register (P2FS, 0x1A24)**

| 0x1A24, R/W | P2 Special Function Select Register, P2FS | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | P27FNC | - | P25FNC | P24FNC | P23FNC | P22FNC | P21FNC | P20FNC |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 0, GPIO
- 1, Special function. P27FNC and P25FNC~P20FNC determines the special functions of P2.7 and P2.5~P2.0. See Table 13-22 for details.

**Table 13-22 Function Configuration of P2**

| P2 Ports | P2FS | | |
|---|---|---|---|
| | Bit | 1 | 0 |
| P2.0 | F20FNC | RXD4 (I), UART4 RX | GPIO |
| P2.1 | F21FNC | TXD4 (O), UART4 TX | |
| P2.2 | F22FNC | RXD5 (I), UART5 RX | |
| P2.3 | F23FNC | TXD5 (O), UART5 TX | |
| P2.4 | F24FNC | INT0 (I), IO Interrupt 0 | |
| P2.5 | F25FNC | INT1 (I), IO Interrupt 1 | |
| P2.7 | F27FNC | SP, Second pulse output | |

# 13.4. P3

The 5 P3 ports are multiplex with special function IOs.

No matter with special function or not, the output and input on IO port are controlled by output enable register and input enable register respectively. Users can use the special function select register to configure the special functions of each IO port.

**Table 13-23 P3 Input Enable Register (P3IE, 0x1A30)**

| 0x1A30, R/W | P3 Input Enable Register, P3IE | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |

| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|---|---|

- 1, to enable input; 0, to disable input

**Table 13-24 P3 Output Enable Register (P3OEN, 0x1A31)**

| 0x1A31, R/W | P3 Output Enable Register, P3OEN | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

- 1, to disable output; 0, to enable output

**Table 13-25 P3 Input Data Register (P3ID, 0x1A32)**

| 0x1A32, R/W | P3 Input Data Register, P3ID | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Table 13-26 P3 Output Data Register (P3OD, 0x1A33)**

| 0x1A33, R/W | P3 Output Data Register, P3OD | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Table 13-27 P3 Special Function Select Register (P3FS, 0x1A34)**

| 0x1A34, R/W | P3 Special Function Select Register, P3FS | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | P35FNC | P34FNC | - | P32FNC | P31FNC | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 0, GPIO
- 1, Special function. P35FNC, P34FNC, P32FNC and P31FNC determine the special functions of P3.5, P3.4, P3.2 and P3.1, see Table 13-28 for details.

**Table 13-28 Function Configuration of P3**

| P3 Ports | P3FS | | |
|---|---|---|---|
| | Bit | 1 | 0 |
| P3.1 | P31FNC | TA1, TimerA Port1 | GPIO |
| P3.2 | P32FNC | TA2, TimerA Port2 | |
| P3.4 | P34FNC | TB1, TimerB Port1 | |
| P3.5 | P35FNC | TB2, TimerB Port2 | |

Users can configure the register P3FS (0x1A34) to use P3 ports for TimerA/B signal input or output which is determined by the registers P3IE (0x1A30) and P3OEN (0x1A31).

Every ports of P3 can be configured as WDTI source. It can be set rising-edge-triggered or falling-edge-triggered. Users can detect the corresponding bits in the register P3IF (0x1A39) for interrupt source.

**Table 13-29 P3 Interrupt Enable Register (P3INTE, 0x1A37)**

| 0x1A37, R/W | P3 Interrupt Enable Register, P3INTE | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 0, to disable interrupt
- 1, to enable interrupt

**Table 13-30 P3 Interrupt Triggering-edge Select Register (P3IES, 0x1A38)**

| 0x1A38, R/W | P3 Interrupt Triggering-edge Select Register, P3IES | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 0, rising-edge triggered interrupt
- 1, falling-edge triggered interrupt

**Table 13-31 P3 Interrupt Flag Register (P3IF, 0x1A39)**

| 0x1A39, R | P3 Interrupt Flag Register, P3IF | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | - | - | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 0, no interrupt is generated
- 1, one or more interrupts are generated

This is a read-only register. Write of the following independent interrupt flag registers can clear flags.

**Table 13-32 P3.1 Interrupt Flag Register (P3IF1, 0x1AA1)**

| 0x1AA1, R/W | P3.1 Interrupt Flag Register, P3IF1 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | P3IF1 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 0, configured by program. This bit cannot be clear automatically via interrupt response.
- 1, an interrupt occurred on P3.1

**Table 13-33 P3.2 Interrupt Flag Register (P3IF2, 0x1AA2)**

| 0x1AA2, R/W | P3.2 Interrupt Flag Register, P3IF2 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | P3IF2 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 0, configured by program. This bit cannot be clear automatically via interrupt response.
- 1, an interrupt occurred on P3.2

**Table 13-34 P3.4 Interrupt Flag Register (P3IF4, 0x1AA4)**

| 0x1AA4, R/W | P3.4 Interrupt Flag Register, P3IF4 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | P3IF4 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 0, configured by program. This bit cannot be clear automatically via interrupt response.
- 1, an interrupt occurred on P3.4

**Table 13-35 P3.5 Interrupt Flag Register (P3IF5, 0x1AA5)**

| 0x1AA5, R/W | P3.5 Interrupt Flag Register, P3IF5 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | P3IF5 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 0, configured by program. This bit cannot be clear automatically via interrupt response.
- 1, an interrupt occurred on P3.5

**Table 13-36 P3.6 Interrupt Flag Register (P3IF6, 0x1AA6)**

| 0x1AA6, R/W | P3.6 Interrupt Flag Register, P3IF6 | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | P3IF6 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 0, configured by program. This bit cannot be clear automatically via interrupt response.
- 1, an interrupt occurred on P3.6

# 13.5. P5

There are 8 P5 ports. They are multiplex with SEG output. Both output and input must be disabled when these pins are used as SEG ports. These pins must be configured to be general digital IOs when they are used as general purpose IOs.

**Table 13-37 P5 Input Enable Register (P5IE, 0x1A50)**

| 0x1A50, R/W | P5 Input Enable Register, P5IE | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 1, to enable input; 0, to disable input

**Table 13-38 P5 Output Enable Register (P5OEN, 0x1A51)**

| 0x1A51, R/W | P5 Output Enable Register, P5OEN | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

- 1, to disable output; 0, to enable output

**Table 13-39 P5 Input Data Register (P5ID, 0x1A52)**

| 0x1A52, R/W | P5 Input Data Register, P5ID | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | | | | | | | | |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Table 13-40 P5 Output Data Register (P5OD, 0x1A53)**

| 0x1A53, R/W | P5 Output Data Register, P5OD | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

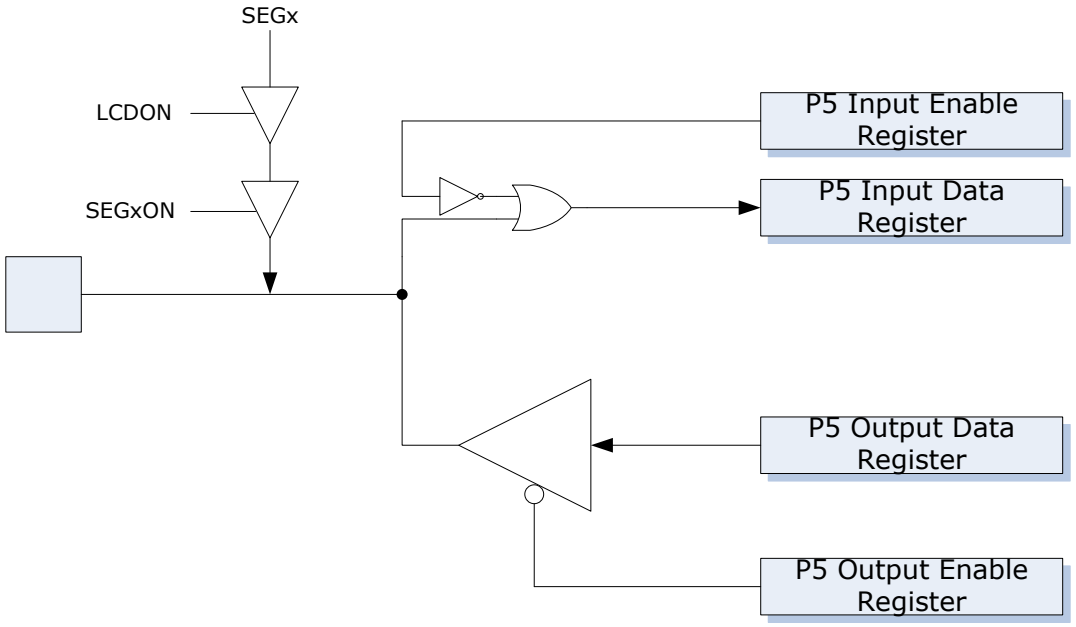



**Figure 13-2 P5 Port Architecture**

# 13.6. P6

There are 8 P6 ports. They are multiplex with and SEG output. Both output and input must be disabled when these pins are used as SEG ports. These pins must be configured to be general digital IOs when they are used as general purpose IOs.

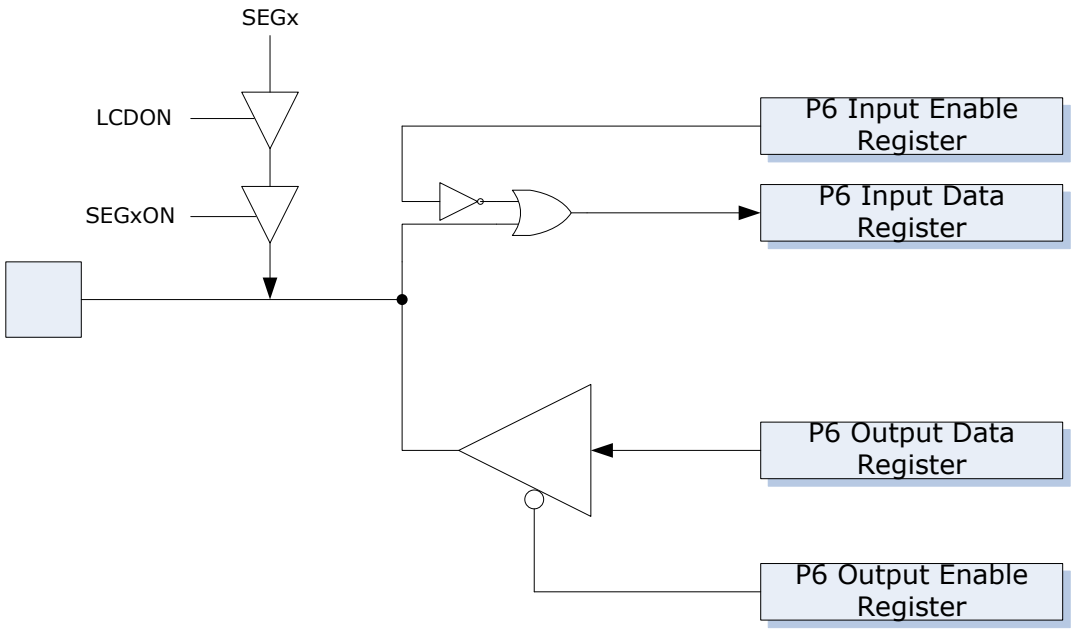



**Figure 13-3 P6 Port Architecture**

**Table 13-41 P6 Input Enable Register (P6IE, 0x1A60)**

| 0x1A60, R/W | P6 Input Enable Register, P6IE | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 1, to enable input; 0, to disable input

**Table 13-42 P6 Output Enable Register (P6OEN, 0x1A61)**

| 0x1A61, R/W | P6 Output Enable Register, P6OEN | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

- 1, to disable output; 0, to enable output

**Table 13-43 P6 Input Data Register (P6ID, 0x1A62)**

| 0x1A62, R/W | P6 Input Data Register, P6ID | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Table 13-44 P6 Output Data Register (P6OD, 0x1A63)**

| 0x1A63, R/W | P6 Output Data Register, P6OD | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

# 13.7. P7

There are 8 P7 ports. They are multiplex with SEG output. Both output and input must be disabled when these pins are used as SEG ports. These pins must be configured to be general digital IOs when they are used as general purpose IOs.

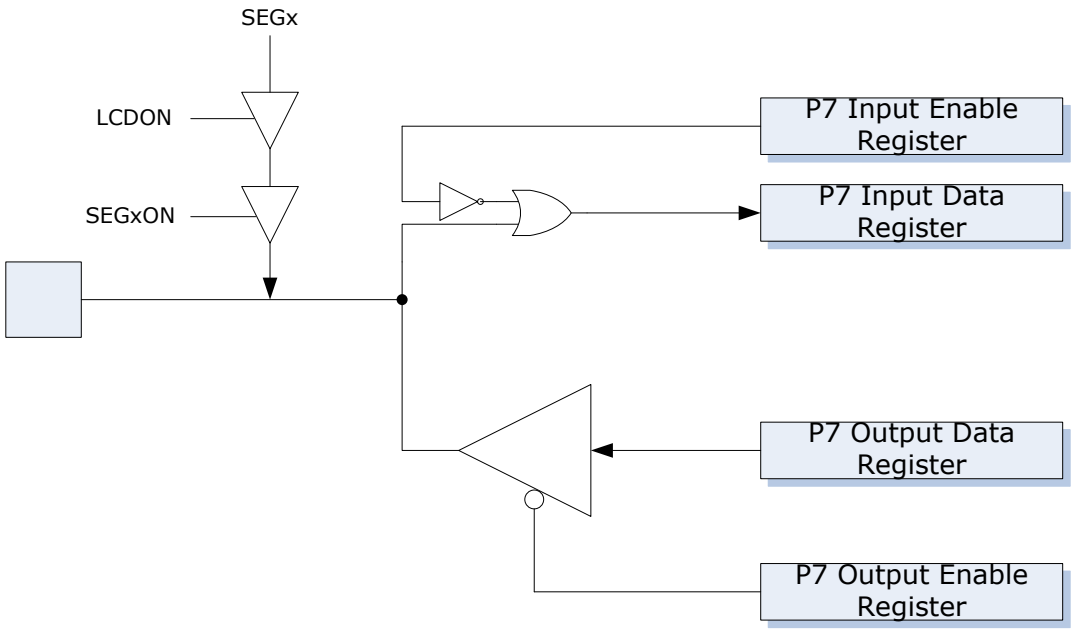



**Figure 13-4 P7 Port Architecture**

**Table 13-45 P7 Input Enable Register (P7IE, 0x1A70)**

| 0x1A70, R/W | P7 Input Enable Register, P7IE | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | - | - | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 1, to enable input; 0, to disable input

**Table 13-46 P7 Output Enable Register (P7OEN, 0x1A71)**

| 0x1A71, R/W | P7 Output Enable Register, P7OEN | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

- 1, to disable output; 0, to enable output

**Table 13-47 P7 Input Data Register (P7ID, 0x1A72)**

| 0x1A72, R/W | P7 Input Data Register, P7ID | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Table 13-48 P7 Output Data Register (P7OD, 0x1A73)**

| 0x1A73, R/W | P7 Output Data Register, P7OD | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

# 13.8. P8

There are 8 P8 ports. Five of them, P8.0~P8.4, are multiplex with SEG output, and three of them, P8.5 ~ P8.7, are multiplex with SEG output and signal input for M Channel. When these pins (P8.0~P8.7) are used as SEG ports, both output and input must be disabled. When these pins (P8.0~P8.7) are used as general-purpose I/O ports, they must be configured to be general digital IOs. When the pins P8.5~P8.7 are used as input ports for M Channel, they should be configured input/output disabled, and the corresponding LCD drivers should be turned off.

**Table 13-49 P8 Input Enable Register (P8IE, 0x1A80)**

| 0x1A80, R/W | P8 Input Enable Register, P8IE | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- 1, to enable input; 0, to disable input

**Table 13-50 P8 Output Enable Register (P8OEN, 0x1A81)**

| 0x1A81, R/W | P8 Output Enable Register, P8OEN | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

- 1, to disable output; 0, to enable output

**Table 13-51 P8 Input Data Register (P8ID, 0x1A82)**

| 0x1A82, R/W | P8 Input Data Register, P8ID | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

**Table 13-52 P8 Output Data Register (P8OD, 0x1A83)**

| 0x1A83, R/W | P8 Output Data Register, P8OD | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | - | - |
| Default Value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

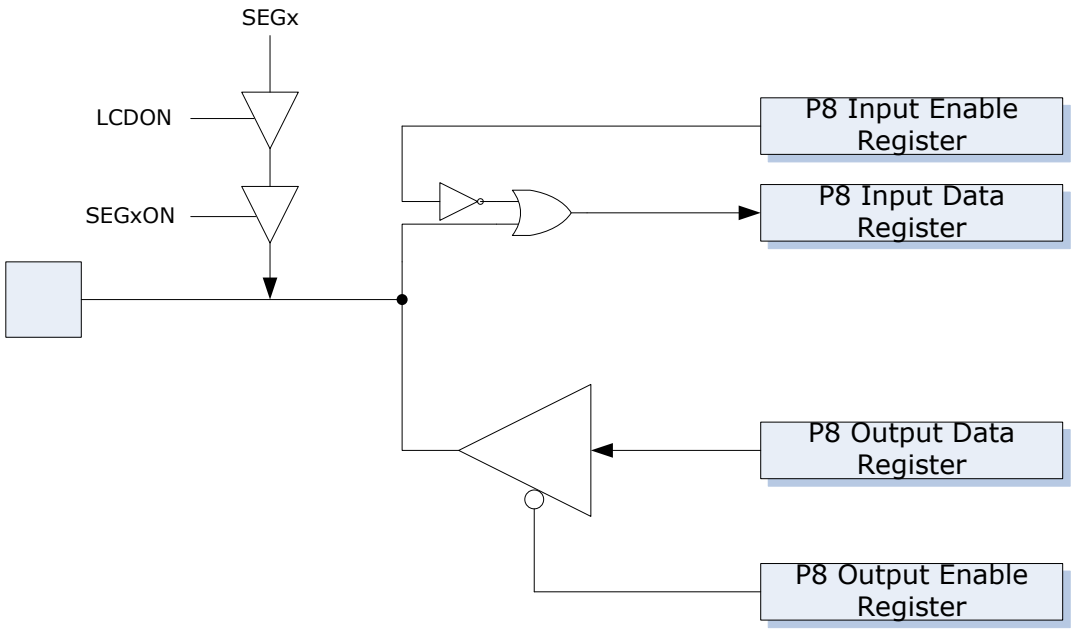



**Figure 13-5 P8 Port Architecture**

# 14. WDT

The Watch Dog Timer (WDT) uses the independent 32.768 kHz RC clock or the 32.768 kHz OSC clock shared with the system, which can be selected via configuring the bit WDCKSEL (Table 14-1) in the register FRQCFG1 (0x1B16). But OSC clock is recommended.

**Table 14-1 Description of The Bit WDCKSEL**

| Bit Field | Function | Description |
|---|---|---|
| WDCKSEL | To select the clock source for WDT | 0, to use RC clock; 1, to use OSC clock |

POR/BOR, RST Pin Reset, and IO/RTC Wake-up Reset can clear the watchdog timer. The watchdog timer runs automatically after reset. When running, if the watchdog timer is not cleared by program in 1.5 seconds, a WDT Reset occurs. If the watchdog timer is still not cleared by program, one reset occurs every 2 seconds. The WDT Reset lasts for 20 ms. WDT stops counting in Sleep or Deep Sleep Mode. After IO/RTC Wake-up Reset, WDT starts running automatically.

When the WDT Reset occurs, the register Wake-up/Reset Reason Query Register (WRS) is configured to 0x01, for software queries. It can be cleared by Recovery Reset.

In debug mode, WDT Reset is masked.

The watchdog timer is cleared by consecutively writing the password for selecting WDT, 0xA5, into the register WDTEN (0x1300), and 0x5B into the register WDTCLR (0x1301) to clear the watchdog timer. Table 14-2 describes the register WDTCLR.

**Table 14-2 WDT Clearing Register (WDTCLR, 0x1301)**

| Address | Register | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
|---|---|---|---|---|---|---|---|---|---|
| 0x1301 | WDTCLR | 0 | 1 | 0 | 1 | 1 | 0 | 1 | WDTCLR |

- When bit7 ~bit1 are set to 0101101, and bit0 (WDTCLR) is set to 1, the watchdog timer is cleared, and no WDT reset occurs. When bit7~ bit1 are set to 0101101, and bit0 (WDTCLR) is set to 0, the watchdog timer works normally.

# 15. Real-Time Clock (RTC)

RTC uses 32768Hz OSC clock or 6.5536MHz PLL clock as clock source. RTC can output second pulse (50% duty cycle, T = 1s) and generate interrupt to MCU every second. There are embedded calibration registers in the RTC which can be used to calibrate time automatically. When the system is in Sleep or Deep Sleep Mode, RTC can provide wake-up reset with intervals of 1 day/1 hour/1 minute/1 second/500ms/250ms/125ms/62.5ms. The timing registers of RTC cannot be reset by any reset.

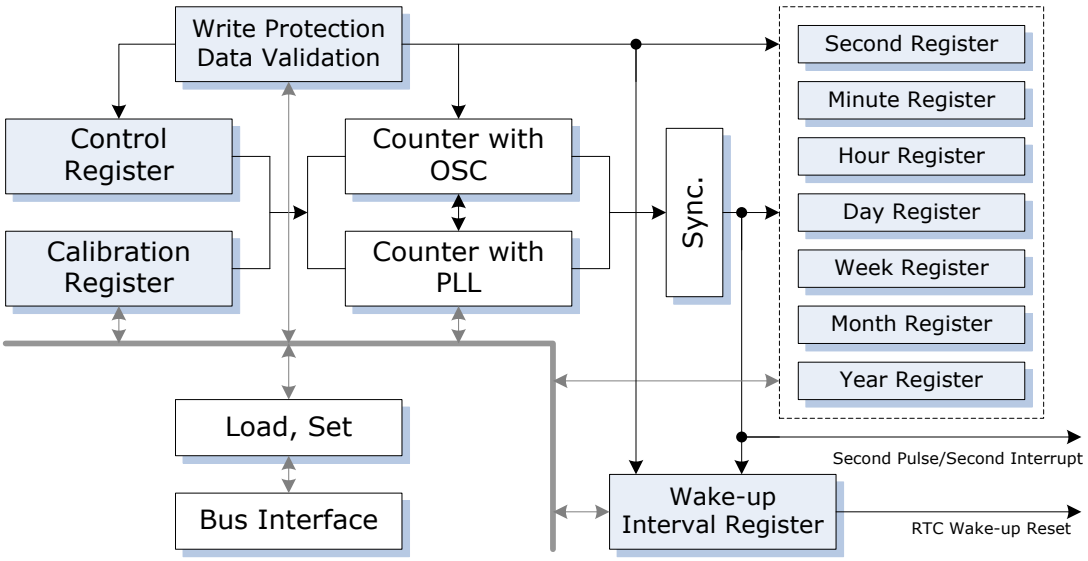



**Figure 15-1 RTC Architecture**

## 15.1. RTC-related Registers

POR/BOR and RST Pin Reset can reset the following registers to the default status: Password Enable Register (RTCPEN, 0x1D0D), RTC Password Register (RTCPWD, 0x1D0A), Calibration Register in PLL Mode (RTCHCH, 0x1D06, and RTCHCL, 0x1D07), Calibration Register in OSC Mode (RTCLCH, 0x1D0C, and RTCLCL, 0x1D0E), RTC Selectable Frequency Time-out Register (RTCSFTO, 0x1D08) and RTC Wake-up Interval Register (INTRTC, 0x1D09). All RTC timing registers cannot be affected by any reset, but can be cleared when the bit RTCRSTn of RTCPWD is set to 0. After the initial power-up, the bit RTCRSTn is set to 0 and RTC do not run until the bit RTCRSTn is set to 1.

**Table 15-1 RTC Password Enable Register (RTCPEN, 0x1D0D)**

| 0x1D0D, R/W | RTC Password Enable Register, RTCPEN | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | bit7 | bit6 | Bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | 0x96 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| Default Value | 0x00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 15-2 RTC Password Register (RTCPWD, 0x1d0A)**

| 0x1d0A, R/W | RTC Password Register, RTCPWD | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | | RTCRSTn | - | - | - | - | - | - | WE |
| Reset RTC | 0x6A/0x6B | 0 | 1 | 1 | 0 | 1 | 0 | 1 | - |
| Restart RTC | 0xEA/0xEB | 1 | 1 | 1 | 0 | 1 | 0 | 1 | - |
| Turn off write protection | 0xD7/0x57 | - | 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| Restore write protection | 0xD6/0x56 | - | 1 | 0 | 1 | 0 | 1 | 1 | 0 |
| Default Value | 0x00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- In V9003/V9103, the registers INTRTC, RTCSFTO, RTC calibration registers, and RTC timing registers are protected from errant write operation. Write 0x96 into the register RTCPEN to enable writing of the register RTCPWD;
- Only when RTCPEN=0x96, and 101011 is written into bit6~bit1 of the register RTCPWD, the write enable bit WE (bit0 of RTCPWD) can be written. Set the bit WE to 1 to disable the write protection on registers INTRTC, RTCSFTO, RTC calibration registers, and RTC timing registers, and configure these registers, and then, set the bit WE to 0 to enable the write protection on these registers, the configured values is loaded into the corresponding registers, and RTC is configured. The configuration of Bit6~bit1 has no effect on read operation of the register RTCPWD;
- Only when RTCPEN=0x96, and 110101 is written into bit6~bit1 of the register RTCPWD, the bit RTCRSTn can be written. When RTCRSTn=0, the RTC is reset, and all RTC related registers are reset. The bit RTCRSTn can be reset by a POR, but not by RST Pin Reset. When a POR occurs, RTCRSTn=1;
- When powered on, RTCRSTn=1, RTC starts timing, and the timing is not affected by any reset until the power is turned off. If the RTC is not configured, it times from a random value; if the RTC is configured, it times from the configured value; if the user write 0x6A/0x6B and 0xEA/0xEB into the register RTCPWD consecutively, the RTC times from 00:00:00.

- When the RTC is reset, users can configure it as follows:
  - Write 0x96 into the register RTCPEN to enable writing the register RTCPWD, and then, write 0xEA/0xEB into RTCPWD to restart RTC;
  - Write 0x96 into the register RTCPEN to enable writing the register RTCPWD, and then, write 0xD7/0x57 into RTCPWD to turn off the write protection on INTRTC, RTCSFTO, RTC calibration registers, and RTC timing registers, and configure them;
  - Write 0x96 into the register RTCPEN to enable writing the register RTCPWD, and then, write 0xD6/0x56 into RTCPWD to restore the write protection on INTRTC, RTCSFTO, RTC calibration registers and RTC timing registers to load the configured values into the corresponding registers, and RTC configuration is completed.
- If users write of INTRTC, RTCSFTO, RTC calibration registers and RTC timing registers when 0xD7/0x57 has not been written into the register RTCPWD, or, error values are written into these registers when 0xD7/0x57 has been written into the register RTCPWD, an RTC illegal interrupt is generated to MCU, and all timing registers hold the previous values. The contents of the timing registers are in binary-coded decimal (BCD) format, so 0xF is valid.

**Table 15-3 Higher Byte of RTC Calibration Register in PLL Mode (RTCHCH, 0x1D06)**

| 0x1D06, R/W | Higher Byte of RTC Calibration Register in PLL Mode, RTCHCH | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | FC14 | FC13 | FC12 | FC11 | FC10 | FC9 | FC8 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- Bit 6 in the register RTCHCH is the sign bit. If the sign bit is 1, the calibration register value is subtracted from the RTC timing registers to slow the PLL clock. If the sign bit is 0, the calibration register value is added into the RTC timing registers to speed up the PLL clock.

**Table 15-4 Lower Byte of RTC Calibration Register in PLL Mode (RTCHCL, 0x1D07)**

| 0x1D07, R/W | Lower Byte of RTC Calibration Register in PLL Mode, RTCHCL | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | FC7 | FC6 | FC5 | FC4 | FC3 | FC2 | FC1 | FC0 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

- By default, bit0 is set to 1, which means no calibration occurs.
- In PLL mode, the RTC can be adjusted over the range ±2500ppm.

**Table 15-5 Higher Byte of RTC Calibration Register in OSC Mode (RTCLCH, 0x1D0C)**

| 0x1D0C, R/W | Higher Byte of RTC Calibration Register in OSC Mode, RTCLCH | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | F5 | F4 | F3 | F2 | F1 | F0 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- Bit5 in the register RTCLCH is the sign bit. If the sign bit is 1, the calibration register value is subtracted from the RTC timing registers to slow the OSC clock. If the sign bit is 0, the calibration register value is added into the RTC timing registers to speed up the OSC clock.

**Table 15-6 Lower Byte of Calibration Register in OSC Mode (RTCLCL, 0x1D0E)**

| 0x1D0E, R/W | Lower Byte of Calibration Register in OSC Mode, RTCLCL | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | F7 | F6 | F5 | F4 | F3 | F2 | F1 | F0 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

- By default, bit0 is set to 1, which means no calibration occurs.

**Table 15-7 RTC Selectable Frequency Time-out Register (RTCSFTO, 0x1D08)**

| 0x1D08, R/W | RTC Selectable Frequency Time-out Register, RTCSFTO | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | - | TSRC | DIVRATE |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- TSRC: to select the clock source. 0, to use 32768Hz OSC clock as the RTC clock source; 1, to use 6.5536 MHz PLL clock as the RTC clock source. The RTC can be set to work in PLL Mode only when PLL is working normally, namely, PLLLCK=1 (Bit4, AFSIG, 0x1B50), and the system clock is set to 6.5536MHz. In this mode, if PLL is turned off (PLLLCK=0, Bit4, AFSIG, 0x1B50), or the frequency of PLL clock is not 6.5536MHz, RTC will be switched to working in OSC Mode automatically.
- DIVRATE: to select the power grid. 0, 50Hz; 1, 60Hz. This bit can only change the configuration of RTC.

**Table 15-8 RTC Wake-up Interval Register (INTRTC, 0x1D09)**

| 0x1D09, R/W | RTC Wake-up Interval Read/Write Register, INTRTC | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | - | - | - | - | - | RTC2 | RTC1 | RTC0 |

| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|---|---|

RTC2, RTC1, and RTC0:

- 000: 1 second
- 001: 1 minute
- 010: 1 hour
- 011: 1 day
- 100: 500 ms
- 101: 250 ms
- 110:125 ms
- 111: 62.5 ms

The time and calendar information is obtained by reading the appropriate register bytes. The contents of the timing registers are in binary-coded decimal (BCD) format, of which, Bit7~bit4 represent the tens digit of the time and calendar, and bit3~bit0 represent the units digit of the time and calendar, for example, write 1000011 into the register RTCSC to represent 43 seconds. The RTC provides seconds, hours, day, week, month, and year information. As such, RTCSC (seconds) and RTCMiC (minutes) range from 0~59, RTCHC (Hour) range from 00~24, RTCDC (Date) is 1~31, RTCMoC (month) is 1~12, RTCYC (year) is 0~99, and RTCWC (Day of Week) is 0~6.

**Table 15-9 Second Setting Register/ Second Register (RTCSC, 0x1D00)**

| 0x1D00, R/W | Second Setting Register/ Second Register, 00~59, RTCSC | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | 0 | S40 | S20 | S10 | S8 | S4 | S2 | S1 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 15-10 Minute Setting Register/ Minute Register (RTCMiC, 0x1D01)**

| 0x1D01, R/W | Minute Setting Register/ Minute Register, 00~59, RTCMiC | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | 0 | M40 | M20 | M10 | M8 | M4 | M2 | M1 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 15-11 Hour Setting Register/ Hour Register (RTCHC, 0x1D02)**

| 0x1D02, R/W | Hour Setting Register/ Hour Register, 00~23, RTCHC | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | 0 | 0 | H20 | H10 | H8 | H4 | H2 | H1 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 15-12 Day Setting Register/ Day Register (RTCDC, 0x1D03)**

| 0x1D03, R/W | Day Setting Register/ Day Register, RTCDC | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | 0 | 0 | D20 | D10 | D8 | D4 | D2 | D1 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Table 15-13 Week Setting Register/ Week Register (RTCWC, 0x1D0B)**

| 0x1D0B, R/W | Week Setting Register/ Week Register, RTCWC | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | 0 | 0 | 0 | 0 | 0 | W3 | W2 | W1 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- The day of the week only can be set, but not detected automatically. Once it is set, RTC can time it automatically. For example, set the date 1st, Jan., 2010, as Friday, and the RTC will determine the day 2nd, Jan., 2010, as Saturday, automatically.
- W3/W2/W1: 000, Sunday; 001, Monday; 010, February; 011, Wednesday; 100, Thursday; 101, Friday; 110, Saturday.

**Table 15-14 Month Setting Register/ Month Register (RTCMoC, 0x1D04)**

| 0x1D04, R/W | Month Setting Register/ Month Register, RTCMoC | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| | 0 | 0 | 0 | M10 | Mo8 | Mo4 | Mo2 | Mo1 |
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Table 15-15 Year Setting Register/ Year Register (RTCYC, 0x1D05)**

| 0x1D05, R/W | Year Setting Register/ Year Register, 2000~2099, RTCYC | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | bit7 | bit6 | bit5 | Bit4 | bit3 | bit2 | Bit1 | bit0 |
| | Y80 | Y40 | Y20 | Y10 | Y8 | Y4 | Y2 | Y1 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Default Value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

- By default, year starts from 2000, so only the tens and units digits of the year needs configuring. For example, write 00010000 into the register to represent the year 2010.

# 15.2. Calibrating RTC

The temperature variation has effect on the actual frequency of the 32768Hz crystal. So, users must calibrate it to reduce the timing error.

## 15.2.1.Temperature Compensation

The following figures illustrate how temperature measurement circuit works.

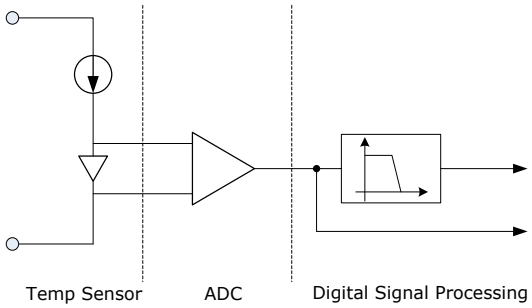



**Figure 15-2 Schematics of Temperature Measurement**

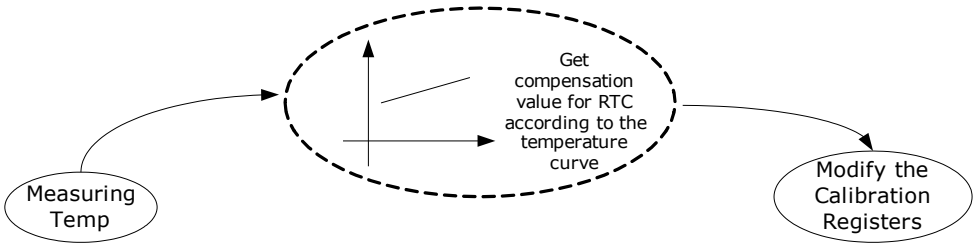



**Figure 15-3 Calibrating RTC over Temperature**

## 15.2.2.Calibrating Sensing Value

Put four energy meters of the same lot into the temperature-controlled cabinet. Adjust the temperature inside the cabinet from-40°C to 85°C, and get the temperature sensor readings. The register values and real temperature are linear related with similar slopes, but the constants are different.

## 15.2.3.Temperature Affecting Crystal Frequency

Put four energy meters of the same lot into the temperature-controlled cabinet. Adjust the temperature inside the cabinet from-40°C to 85°C, and meter the frequency of the second pulse via the pin SP (P2.7) when the LSBs of registers RTCHCL and RTCLCL are set to ones, and then, obtain the crystal frequency via the formula (second pulse frequency × 32768); or users can configure the P1.7 via the registers P1FSEL (0x1A15) and P1FS (0x1A14) for OSC clock output to meter the crystal frequency. The crystal frequency is parabolic related to the temperature. Users can calculate the calibration value over the temperature variation, make a table, and then use the table look-up to obtain the calibration value.

1. Set the crystal frequency reference, for example, 32768Hz.
2. Compare the measured crystal frequency "x" to the reference, and calculate the calibration value "y":

   In PLL mode, the calibration resolution is $\frac{1}{6553600} \times 1000000 = 0.153$ ppm; when the measured crystal frequency is x, $\frac{x - 32768}{32768} \times 1000000$ ppm must be calibrated; so the calibration value can be calculated by the formula:

   $$y = \frac{x - 32768}{32768} \times 1000000 \ / (\frac{1}{6553600} \times 1000000) = (x - 32768) \times 200 .$$

3. Write the value of "y" into the higher and lower bytes of the calibration register in PLL Mode.

## 15.2.4.Testing

1. Read the temperature, and output the second pulses;
2. From -40°C to 85°C, read the temperature and second pulses frequency of the meters;
3. Test once per 5°C. The measurement temperature of the meter is identical to the real temperature, and after crystal frequency compensation over the temperature variation, the offset of the second pulses is in the range -1.1~ +2.1ppm, meeting the requirement of ±5ppm.

## 15.2.5.Quick Steps to Get the Calibrated Data

1. Turn the system clock to use the 6.5536 MHz PLL clock;
2. Set the bit TSRC (Bit1, RTCSFTO) to 1 to start RTC calibration in PLL Mode;

3. Write the higher byte of the calibrating value into the register RTCHCH, and the lower byte into the register RTCHCL, which improve the accuracy of the second pulse. The output second pulse is calibrated.

4. Write $Calc_{PLL}$ into the RTC calibration register in PLL Mode (without the sign bit). Use the following equation to get $Calc_{OSC}$ for the RTC calibration register with OSC

$$Calc_{OSC} = \frac{Calc_{PLL} \times 30}{200} + \frac{17}{20}$$

5. Write $Calc_{OSC}$ and sign bit into the registers RTCLCH and RTCLCL, and write 0 into the bit TSRC of the register RTCSFTO to cancel the calibration register in PLL Mode.

When working in the normal mode, OSC clock is selected as the clock source, and according to the readings of the registers RTCLCH and RTCLCL, RTC is calibrated automatically once per 30 seconds. The calibration precision is 30.5ppm/30s and the calibration is over a range of ±8332ppm (±12min/day). When calibration in PLL Mode is enabled, calibration occurs once per second. The calibration precision is 0.153ppm/s and the calibration is over a range of ±2500ppm (±216 seconds/day). Use quick steps to calibrate RTC can improve the precision of RTC calibration.

Both bit 6 of the register RTCHCH and bit 5 of the register RTCLCH are sign bits. If the sign bit is 1, the calibration register value is subtracted from the RTC timing registers to slow down the OSC or PLL clock. If the sign bit is 0, the calibration register value is added into the RTC timing registers to speed up the OSC or PLL clock.

## 15.2.6.Normal Steps to Get the Calibrated Data

1. Calibrating in PLL Mode

   1.1 Use 6.5536MHz PLL clock as the system clock source;

   1.2 Write 1 into the bit TSRC (Bit1) of the register RTCSFTO to turn on the calibration in PLL Mode;

   1.3 Write the higher byte of the calibrating values into the register RTCHCH, and the lower byte into the register RTCHCL, to improve the accuracy of the second pulses. The output second pulses are calibrated.

2. Calibrating in OSC Mode

   1.1 Write 0 into the bit TSRC (bit1) of the register RTCSFTO to turn on RTC calibration in OSC Mode.

1.2 Adjust the values of the registers RTCLCL and RTCLCH to improve the accuracy of the output second pulses. The second signals are calibrated once per 30 seconds.

1.3 The readings of the registers RTCLCH and RTCLCL are the calibrated values $Calc_{OSC}$ (without the sign bit).

## 15.2.7.Calibration Examples

When calibrating the RTC, set the period of second pulse to T, and use the following equation to calculate the calibrating values (the results are represented in 2'complementary code):

Calibration in PLL Mode: $Calc_{PLL} = (\frac{1}{T} - 1) \times 6553600 + 1$

For example, to slow down by 1000ppm:

1. Calculate $-\frac{1000}{0.153} + 1 = -6535$, convert the absolute value of it to HEX format and get 0x1987;
2. Take 2'complement, 0x8000 - 0x1987 = 0x6679.

Then, RTCHCH is set to 0x66, and RTCHCL is set to 0x79.

To speed up by 1000ppm: calculate $\frac{1000}{0.153} + 1 = 6537$, convert it to HEX format and get 0x1989.

Then, RTCHCH is set to 0x19, and RTCHCL is set to 0x89.

Calibration in OSC Mode: $Calc_{OSC} = (\frac{1}{T} - 1) \times 32768 \times 30 + 1$

For example, to slow down by 1000ppm:

1. Calculate $(-\frac{1000}{30.5} + 1) \times 30 = -953.6$, convert the absolute value of it to HEX format and get 0x03B9;
2. Take 2'complement, 0x4000 - 0x03B9 = 0x3C46.

Then, RTCLCH is set to 0x3C, and RTCLCL is set to 0x46.

To speed up by 1000ppm: calculate $(\frac{1000}{30.5} + 1) \times 30 = 1013.6$, convert it to HEX format and get 0x03F5.

Then, RTCLCH is set to 0x03, and RTCLCL is set to 0xF5.

# 15.3. RTC with PLL Clock

When 6.5536 MHz PLL clock is selected as the clock source of the system clock, write 1 into the bit TSRC (bit 1) of the register RTCSFTO to turn on RTC calibration in PLL Mode, and set RTC work with PLL clock. When working with PLL clock, the RTC uses 6.5536MHz PLL clock as the clock source, and RTC is calibrated once per second according to the values of the registers RTCHCH and RTCHCL, with 0.153ppm calibration accuracy, and over the range of ±2500ppm (±216s/day).

When 32768Hz OSC clock is used as the system clock source, writing 1 into the bit TSRC (bit 1) is invalid. Write 0 into the bit TSRC to cancel the RTC with PLL clock, and then, switch the system clock source from PLL clock into OSC clock. If the system clock source is switched from PLL clock into OSC clock without writing 0 into the bit TSRC, or PLL clock is turned off directly, the bit TSRC is cleared automatically, and calibration in PLL Mode is canceled automatically.

# 16. Appendix

1. LDO33 Output and External Voltage Signal Input

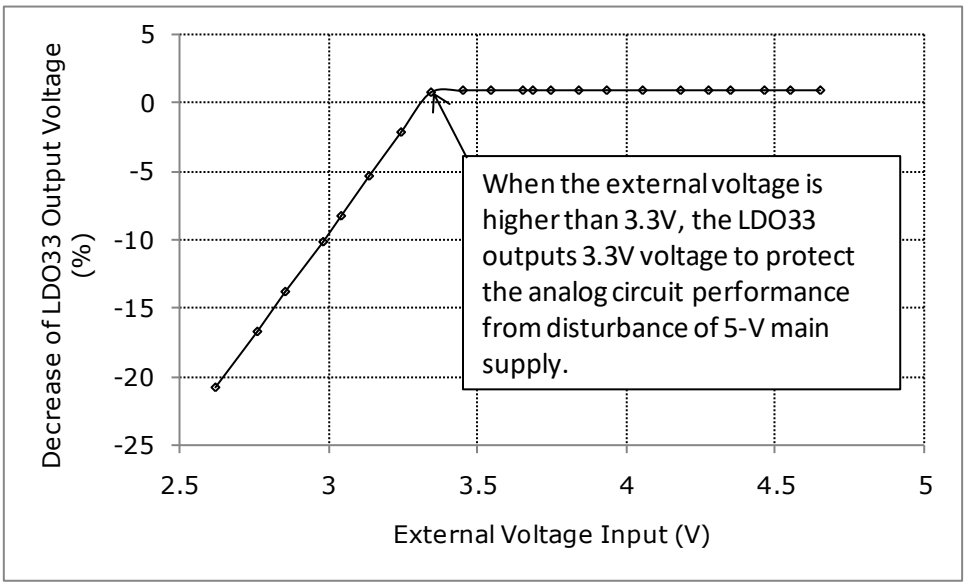



**Figure 16-1 LDO33 Output and External Voltage Signal Input**

2. LDO33 output and the Load Current

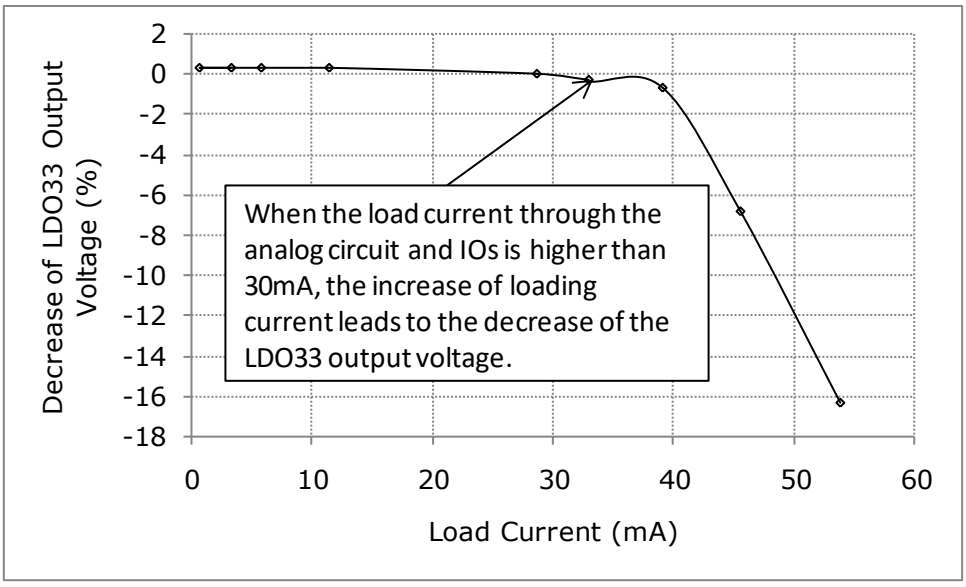



**Figure 16-2 LDO33 output and the Load Current**

3. LDO25 Output and the Load Current

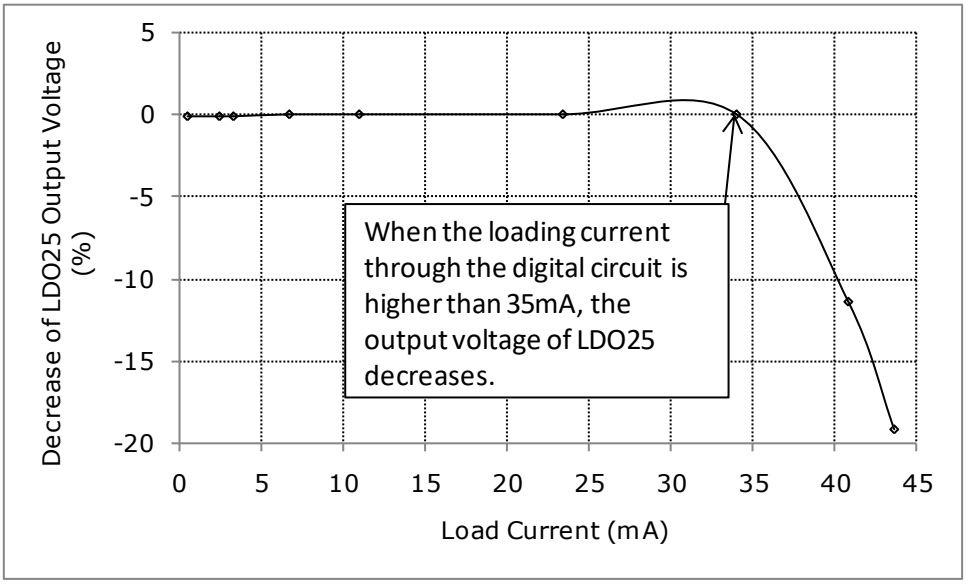



**Figure 16-3 LDO25 Output and the Load Current**

4. Relationship of Reference Voltage Variation and Power Consumption

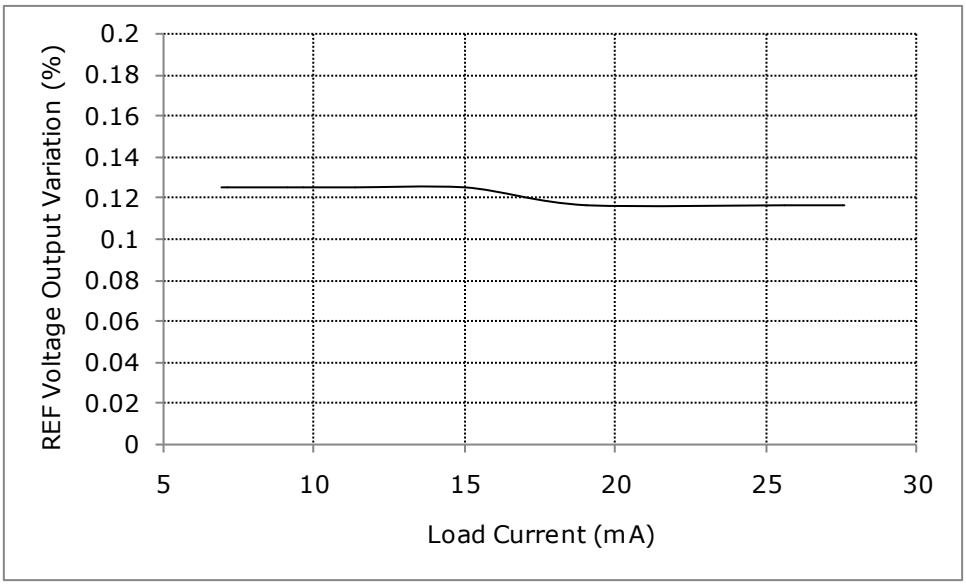



**Figure 16-4 Relationship of Reference Voltage Variation and Power Consumption**

5. Relationship of Reference Voltage Variation and Temperature

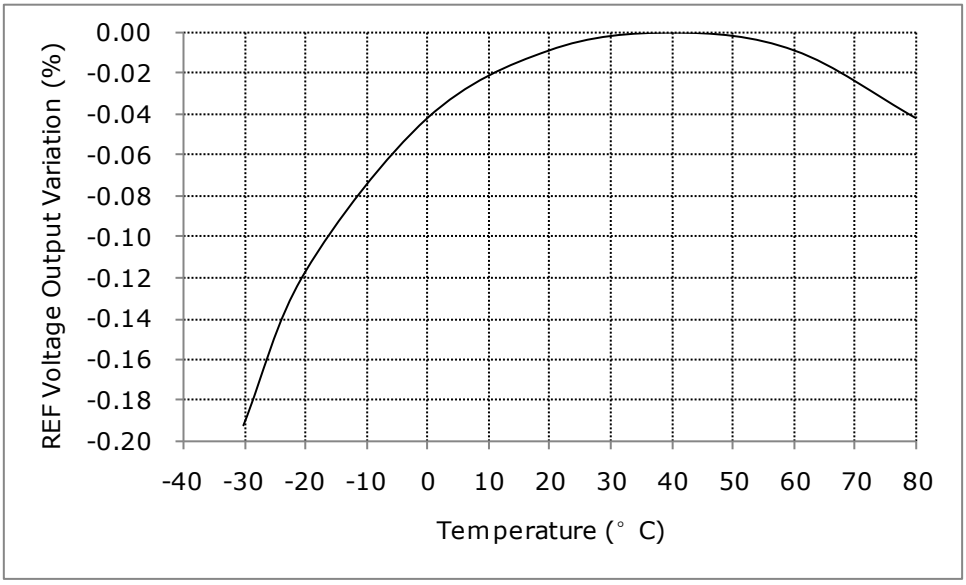



**Figure 16-5 Relationship of Reference Voltage Variation and Temperature**

6. M Channel for External Sources Measurement Divided by the Resistor-division Network

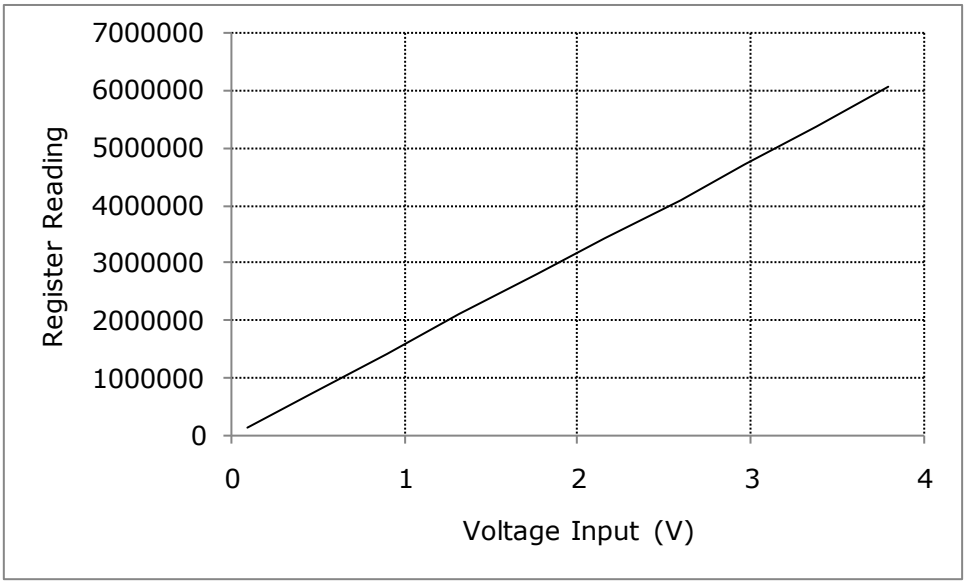



**Figure 16-6 M Channel for External Sources Measurement Divided by the Resistor-division Network**

7. M Channel for External Sources Measurement without Resistor-division Network

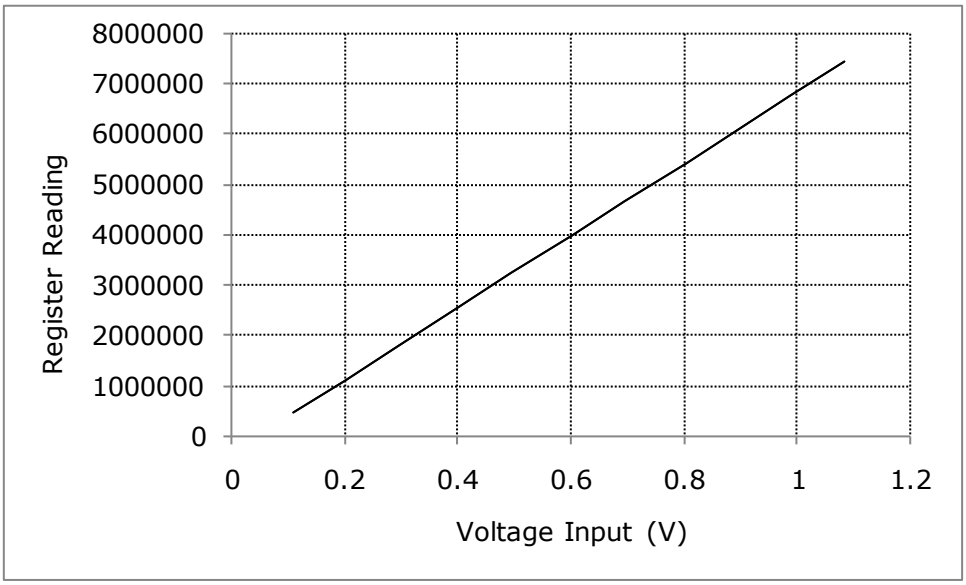
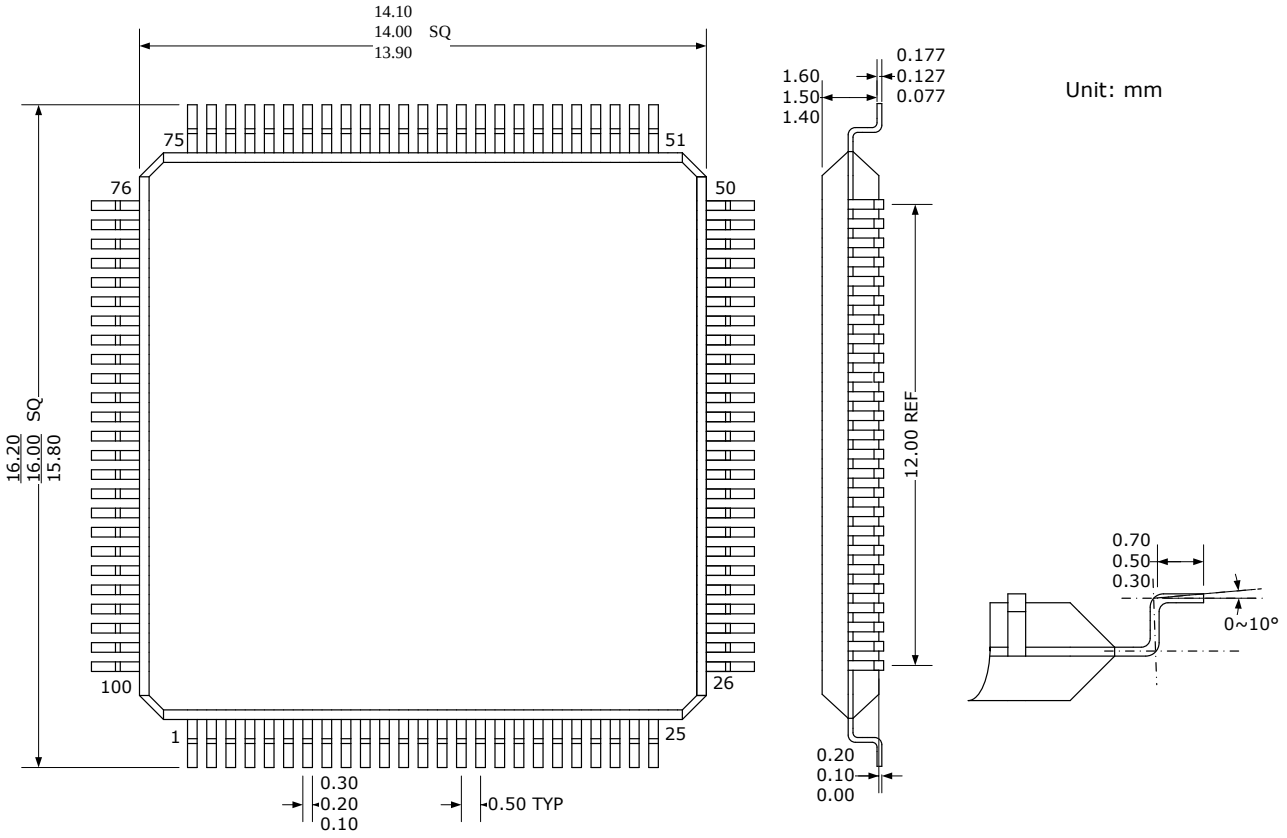



**Figure 16-7 M Channel for External Sources Measurement without Resistor-division Network**

# 17. Outline Dimension

## 17.1. Outline Dimension_V9003



# 17.2. Outline Dimension_V9103