



复旦微电子

# ***FM33A0xx 系列***

## ***ARM 固件函数库用户手册***



## 前言

本手册适用于上海复旦微电子集团推出的 FM33A0xx 系列 32 位 ARM 的编程, 本手册及所提供的函数库仅供参考。

固函数库持续完善中, 欢迎前往开发者论坛下载最新版本:

<http://www.fmdevelopers.com.cn>



# 目录

<b>FM33A0XX 系列</b> .....	<b>1</b>
<b>1 工程使用说明</b> .....	<b>30</b>
<b>2 GPIO</b> .....	<b>31</b>
2.1 GPIO 寄存器结构: .....	31
2.2 GPIO 函数库: .....	33
2.2.1 GPIO_Init.....	34
2.2.2 GPIO_ALL_Deinit.....	34
2.2.3 GPIO_Deinit.....	35
2.2.4 GPIOx_Deinit.....	35
2.2.5 GPIO_ReadInputDataBit.....	35
2.2.6 GPIO_ReadOutputDataBit.....	35
2.2.7 GPIO_ReadInputData.....	36
2.2.8 GPIOx_DIN_Read.....	36
2.2.9 GPIO_ReadOutputData.....	36
2.2.10 GPIOx_DO_Read.....	36
2.2.11 GPIOx_DO_Write.....	37
2.2.12 GPIO_SetBits.....	37
2.2.13 GPIOx_DSET_Write.....	37
2.2.14 GPIO_ResetBits.....	37
2.2.15 GPIOx_DRESET_Write.....	38
2.2.16 GPIO_ToggleBits.....	38
2.2.17 GPIO_Get_InitPara.....	38
2.2.18 GPIO_FOUTSEL_FOUTSEL_Set.....	39
2.2.19 GPIO_FOUTSEL_FOUTSEL_Get.....	39
2.2.20 GPIO_HDSEL_PG6HDEN_Setable.....	40
2.2.21 GPIO_HDSEL_PE2HDEN_Setable.....	40
2.2.22 GPIO_HDSEL_PG6HDEN_Getable.....	41
2.2.23 GPIO_HDSEL_PE2HDEN_Getable.....	41
2.2.24 GPIO_ANASEL_PE4ANS_Set.....	41
2.2.25 GPIO_ANASEL_PE4ANS_Get.....	41
2.2.26 GPIO_ANASEL_PE3ANS_Set.....	42
2.2.27 GPIO_ANASEL_PE3ANS_Get.....	42
2.2.28 GPIO_ANASEL_PC15ANS_Set.....	42
2.2.29 GPIO_ANASEL_PC15ANS_Get.....	43
2.2.30 GPIO_ANASEL_PC14ANS_Set.....	43



2.2.31	<i>GPIO_ANASEL_PC14ANS_Get</i> .....	43
2.2.32	<i>GPIO_ANASEL_PC13ANS_Set</i> .....	43
2.2.33	<i>GPIO_ANASEL_PC13ANS_Get</i> .....	44
2.2.34	<i>GPIO_ANASEL_PC12ANS_Set</i> .....	44
2.2.35	<i>GPIO_ANASEL_PC12ANS_Get</i> .....	44
2.2.36	<i>GPIO_EXTI_Init</i> .....	45
2.2.37	<i>GPIO_EXTI_Close</i> .....	45
2.2.38	<i>GPIO_EXTI_Select_Pin</i> .....	45
2.2.39	<i>GPIO_EXTI_Select_Edge</i> .....	46
2.2.40	<i>GPIO_EXTI_EXTIxIF_ClrEx</i> .....	46
2.2.41	<i>GPIO_EXTI_EXTIxIF_ChkEx</i> .....	46
2.2.42	<i>GPIO_IODF_SetableEx</i> .....	47
2.2.43	<i>GPIO_PINWKEN_SetableEx</i> .....	47
<b>3</b>	<b>RCC</b> .....	<b>48</b>
3.1	RCC 寄存器结构 .....	48
3.2	RCC 库函数 .....	48
3.2.1	<i>RCC_SysClk_Init</i> .....	49
3.2.2	<i>RCC_RCHF_Init</i> .....	50
3.2.3	<i>RCC_PLL_Init</i> .....	51
3.2.4	<i>RCC_Deinit</i> .....	51
3.2.5	<i>RCC_PERCLK_SetableEx</i> .....	51
3.2.6	<i>RCC_Init_RCHF_Trim</i> .....	52
3.2.7	<i>RCC_GetClocksFreq</i> .....	52
3.2.8	<i>RCC_SYSCLKSEL_SYSCLKSEL_Set</i> .....	52
3.2.9	<i>RCC_SYSCLKSEL_SYSCLKSEL_Get</i> .....	53
3.2.10	<i>RCC_RCHFCON_FSEL_Set</i> .....	53
3.2.11	<i>RCC_RCHFCON_FSEL_Get</i> .....	54
3.2.12	<i>RCC_RCHFCON_RCHFEN_Setable</i> .....	54
3.2.13	<i>RCC_RCHFCON_RCHFEN_Getable</i> .....	54
3.2.14	<i>RCC_RCHFTRIM_Write</i> .....	54
3.2.15	<i>RCC_RCHFTRIM_Read</i> .....	55
3.2.16	<i>RCC_SYSCLKSEL_LPM_RCLP_OFF_Setable</i> .....	55
3.2.17	<i>RCC_SYSCLKSEL_LPM_RCLP_OFF_Getable</i> .....	55
3.2.18	<i>RCC_RCLPTRIM_Write</i> .....	56
3.2.19	<i>RCC_RCLPTRIM_Read</i> .....	56
3.2.20	<i>RCC_RCLPCON_RCLP_EN_B_Chk</i> .....	56
3.2.21	<i>RCC_XTLFIPW_XTLFIPW_Set</i> .....	56
3.2.22	<i>RCC_XTLFIPW_XTLFIPW_Get</i> .....	57
3.2.23	<i>RCC_PLLCON_PLLDB_Set</i> .....	57



3.2.24	<i>RCC_PLLCON_PLLDB_Get</i> .....	57
3.2.25	<i>RCC_PLLCON_PLLOSEL_Set</i> .....	58
3.2.26	<i>RCC_PLLCON_PLLOSEL_Get</i> .....	58
3.2.27	<i>RCC_PLLCON_PLLINSEL_Set</i> .....	58
3.2.28	<i>RCC_PLLCON_PLLINSEL_Get</i> .....	59
3.2.29	<i>RCC_PLLCON_PLEN_Setable</i> .....	59
3.2.30	<i>RCC_PLLCON_PLEN_Getable</i> .....	59
3.2.31	<i>RCC_SYSCLKSEL_AHBPRES_Set</i> .....	59
3.2.32	<i>RCC_SYSCLKSEL_AHBPRES_Get</i> .....	60
3.2.33	<i>RCC_MPRIL_MPRIL_Set</i> .....	60
3.2.34	<i>RCC_MPRIL_MPRIL_Get</i> .....	61
3.2.35	<i>RCC_SYSCLKSEL_APBPRES_Set</i> .....	61
3.2.36	<i>RCC_SYSCLKSEL_APBPRES_Get</i> .....	61
3.2.37	<i>RCC_SYSCLKSEL_EXTICKSEL_Set</i> .....	62
3.2.38	<i>RCC_SYSCLKSEL_EXTICKSEL_Get</i> .....	62
3.2.39	<i>RCC_SYSCLKSEL_SLP_ENEXTI_Setable</i> .....	62
3.2.40	<i>RCC_SYSCLKSEL_SLP_ENEXTI_Getable</i> .....	62
3.2.41	<i>RCC_PERCLKCON2_ADCCCKSEL_Set</i> .....	63
3.2.42	<i>RCC_PERCLKCON2_ADCCCKSEL_Get</i> .....	63
<b>4</b>	<b>PMU</b> .....	<b>65</b>
4.1	PMU 寄存器结构.....	65
4.2	PMU 函数库.....	65
4.2.1	<i>PMU_SleepCfg_Init</i> .....	66
4.2.2	<i>PMU_Deinit</i> .....	67
4.2.3	<i>PMU_LPMCFG_PMOD_Set</i> .....	67
4.2.4	<i>PMU_LPMCFG_PMOD_Get</i> .....	67
4.2.5	<i>PMU_LPMCFG_CVS_Setable</i> .....	67
4.2.6	<i>PMU_LPMCFG_CVS_Getable</i> .....	68
4.2.7	<i>PMU_LPREIE_LPREIE_Setable</i> .....	68
4.2.8	<i>PMU_LPREIE_LPREIE_Getable</i> .....	68
4.2.9	<i>PMU_LPREIF_LPREIF_Clr</i> .....	69
4.2.10	<i>PMU_LPREIF_LPREIF_Chk</i> .....	69
4.2.11	<i>PMU_LPMCFG_SLPDP_Set</i> .....	69
4.2.12	<i>PMU_LPMCFG_SLPDP_Get</i> .....	69
4.2.13	<i>PMU_LPMCFG_DSLPRAM_EXT_Set</i> .....	70
4.2.14	<i>PMU_LPMCFG_DSLPRAM_EXT_Get</i> .....	70
4.2.15	<i>PMU_LPREIE_SLPEIE_Setable</i> .....	70
4.2.16	<i>PMU_LPREIE_SLPEIE_Getable</i> .....	71
4.2.17	<i>PMU_LPREIF_SLPEIF_Clr</i> .....	71



4.2.18	<i>PMU_LPREIF_SLPEIF_Chk</i> .....	71
4.2.19	<i>PMU_WKDLYCON_TRHR_Set</i> .....	71
4.2.20	<i>PMU_WKDLYCON_TRHR_Get</i> .....	72
4.2.21	<i>PMU_WKPFLAG_DBGWKF_Clr</i> .....	72
4.2.22	<i>PMU_WKPFLAG_DBGWKF_Chk</i> .....	72
4.2.23	<i>PMU_WKPFLAG_WKPxF_ClrEx</i> .....	73
4.2.24	<i>PMU_WKPFLAG_WKPxF_ChkEx</i> .....	73
4.2.25	<i>PMU_LPMCFCG_XTOFF_Set</i> .....	73
4.2.26	<i>PMU_LPMCFCG_XTOFF_Get</i> .....	74
4.2.27	<i>PMU_LPMCFCG_LDO15EN_Chk</i> .....	74
<b>5</b>	<b>FLASH</b> .....	<b>75</b>
5.1	FLASH 寄存器结构.....	75
5.2	FLASH 函数库.....	75
5.2.1	<i>FLASH_Deinit</i> .....	77
5.2.2	<i>FLASH_FLSRDCON_WAIT_Set</i> .....	77
5.2.3	<i>FLASH_FLSRDCON_WAIT_Get</i> .....	77
5.2.4	<i>FLASH_Prog_SingleByte</i> .....	77
5.2.5	<i>FLASH_Prog_ByteString</i> .....	78
5.2.6	<i>FLASH_Erase_Sector</i> .....	78
5.2.7	<i>FLASH_OPTBR_DBGCFGEN_Chk</i> .....	78
5.2.8	<i>FLASH_OPTBR_RAMBOOT_Chk</i> .....	79
5.2.9	<i>FLASH_OPTBR_ACLOCKEN_Chk</i> .....	79
5.2.10	<i>FLASH_OPTBR_DBRDPEN_Chk</i> .....	79
5.2.11	<i>FLASH_EPCON_ERTYPE_Set</i> .....	79
5.2.12	<i>FLASH_EPCON_ERTYPE_Get</i> .....	80
5.2.13	<i>FLASH_EPCON_PREQ_Set</i> .....	80
5.2.14	<i>FLASH_EPCON_PREQ_Get</i> .....	80
5.2.15	<i>FLASH_EPCON_EREQ_Set</i> .....	81
5.2.16	<i>FLASH_EPCON_EREQ_Get</i> .....	81
5.2.17	<i>FLASH_FLSIE_AUTHIE_Setable</i> .....	81
5.2.18	<i>FLASH_FLSIE_AUTHIE_Getable</i> .....	82
5.2.19	<i>FLASH_FLSIE_KEYIE_Setable</i> .....	82
5.2.20	<i>FLASH_FLSIE_KEYIE_Getable</i> .....	82
5.2.21	<i>FLASH_FLSIE_CKIE_Setable</i> .....	82
5.2.22	<i>FLASH_FLSIE_CKIE_Getable</i> .....	83
5.2.23	<i>FLASH_FLSIE_PRDIE_Setable</i> .....	83
5.2.24	<i>FLASH_FLSIE_PRDIE_Getable</i> .....	83
5.2.25	<i>FLASH_FLSIE_ERDIE_Setable</i> .....	84
5.2.26	<i>FLASH_FLSIE_ERDIE_Getable</i> .....	84



5.2.27	FLASH_FLSIF_AUTHIF_Clr.....	84
5.2.28	FLASH_FLSIF_AUTHIF_Chk.....	84
5.2.29	FLASH_FLSIF_KEYIF_Clr.....	85
5.2.30	FLASH_FLSIF_KEYIF_Chk.....	85
5.2.31	FLASH_FLSIF_CKIF_Clr.....	85
5.2.32	FLASH_FLSIF_CKIF_Chk.....	85
5.2.33	FLASH_FLSIF_PRDIF_Clr.....	86
5.2.34	FLASH_FLSIF_PRDIF_Chk.....	86
5.2.35	FLASH_FLSIF_ERDIF_Clr.....	86
5.2.36	FLASH_FLSIF_ERDIF_Chk.....	87
5.2.37	FLASH_FLSKEY_Write.....	87
5.2.38	FLASH_ACLOCK1_Write.....	87
5.2.39	FLASH_ACLOCK1_Read.....	87
5.2.40	FLASH_ACLOCK2_Write.....	88
5.2.41	FLASH_ACLOCK2_Read.....	88
<b>6</b>	<b>IWDT.....</b>	<b>89</b>
6.1	IWDT 寄存器结构.....	89
6.2	IWDT 函数库.....	89
6.2.1	IWDT_Deinit.....	89
6.2.2	IWDT_Clr.....	89
6.2.3	IWDT_IWDTSERV_Write.....	90
6.2.4	IWDT_IWDTCNT_Read.....	90
6.2.5	IWDT_IWDTCFG_IWDTOVP_Set.....	90
6.2.6	IWDT_IWDTCFG_IWDTOVP_Get.....	90
6.2.7	IWDT_IWDTCFG_IWDTSLP4096S_Setable.....	91
6.2.8	IWDT_IWDTCFG_IWDTSLP4096S_Getable.....	91
<b>7</b>	<b>WWDT.....</b>	<b>92</b>
7.1	WWDT 寄存器结构.....	92
7.2	WWDT 函数库.....	92
7.2.1	WWDT_Deinit.....	92
7.2.2	WWDT_Start.....	93
7.2.3	WWDT_Clr.....	93
7.2.4	WWDT_WWDTCON_Write.....	93
7.2.5	WWDT_WWDTCNT_Read.....	93
7.2.6	WWDT_WWDTCFG_WWDTCFG_Set.....	94
7.2.7	WWDT_WWDTCFG_WWDTCFG_Get.....	94
7.2.8	WWDT_WWDTDIV_Read.....	94
7.2.9	WWDT_WWDTIE_WWDTIE_Setable.....	95



7.2.10	<i>WWDT_WWDTIE_WWDTIE_Getable</i> .....	95
7.2.11	<i>WWDT_WWDTIF_WWDTIF_Clr</i> .....	95
7.2.12	<i>WWDT_WWDTIF_WWDTIF_Chk</i> .....	96
<b>8</b>	<b>SCU</b> .....	<b>97</b>
8.1	SCU 寄存器结构 .....	97
8.2	SCU 函数库 .....	97
8.2.1	<i>SCU_Deinit</i> .....	98
8.2.2	<i>SCU_MCUDBGCR_DBG_ET1_STOP_Setable</i> .....	98
8.2.3	<i>SCU_MCUDBGCR_DBG_ET1_STOP_Getable</i> .....	98
8.2.4	<i>SCU_MCUDBGCR_DBG_ET2_STOP_Setable</i> .....	99
8.2.5	<i>SCU_MCUDBGCR_DBG_ET2_STOP_Getable</i> .....	99
8.2.6	<i>SCU_MCUDBGCR_DBG_ET3_STOP_Setable</i> .....	99
8.2.7	<i>SCU_MCUDBGCR_DBG_ET3_STOP_Getable</i> .....	100
8.2.8	<i>SCU_MCUDBGCR_DBG_ET4_STOP_Setable</i> .....	100
8.2.9	<i>SCU_MCUDBGCR_DBG_ET4_STOP_Getable</i> .....	100
8.2.10	<i>SCU_MCUDBGCR_DBG_BT1_STOP_Setable</i> .....	100
8.2.11	<i>SCU_MCUDBGCR_DBG_BT1_STOP_Getable</i> .....	101
8.2.12	<i>SCU_MCUDBGCR_DBG_BT2_STOP_Setable</i> .....	101
8.2.13	<i>SCU_MCUDBGCR_DBG_BT2_STOP_Getable</i> .....	101
8.2.14	<i>SCU_MCUDBGCR_DBG_WWDT_STOP_Setable</i> .....	102
8.2.15	<i>SCU_MCUDBGCR_DBG_WWDT_STOP_Getable</i> .....	102
8.2.16	<i>SCU_MCUDBGCR_DBG_IWDT_STOP_Setable</i> .....	102
8.2.17	<i>SCU_MCUDBGCR_DBG_IWDT_STOP_Getable</i> .....	102
8.2.18	<i>SCU_HDFFLAG_HDF_REQUEST_FLAG_Clr</i> .....	103
8.2.19	<i>SCU_HDFFLAG_HDF_REQUEST_FLAG_Chk</i> .....	103
8.2.20	<i>SCU_HDFFLAG_DABORT_ADDR_FLAG_Clr</i> .....	103
8.2.21	<i>SCU_HDFFLAG_DABORT_ADDR_FLAG_Chk</i> .....	104
8.2.22	<i>SCU_HDFFLAG_DABORT_RESP_FLAG_Clr</i> .....	104
8.2.23	<i>SCU_HDFFLAG_DABORT_RESP_FLAG_Chk</i> .....	104
8.2.24	<i>SCU_HDFFLAG_SVCUNDEF_FLAG_Clr</i> .....	104
8.2.25	<i>SCU_HDFFLAG_SVCUNDEF_FLAG_Chk</i> .....	105
8.2.26	<i>SCU_HDFFLAG_BKPT_FLAG_Clr</i> .....	105
8.2.27	<i>SCU_HDFFLAG_BKPT_FLAG_Chk</i> .....	105
8.2.28	<i>SCU_HDFFLAG_TBIT_FLAG_Clr</i> .....	105
8.2.29	<i>SCU_HDFFLAG_TBIT_FLAG_Chk</i> .....	106
8.2.30	<i>SCU_HDFFLAG_SPECIAL_OP_FLAG_Clr</i> .....	106
8.2.31	<i>SCU_HDFFLAG_SPECIAL_OP_FLAG_Chk</i> .....	106
<b>9</b>	<b>ANAC</b> .....	<b>107</b>





9.1	ANAC 寄存器结构.....	107
9.2	ANAC 库函数.....	107
9.2.1	ANAC_Deinit.....	110
9.2.2	ANAC_PDRCON_PDRCFG_Set.....	110
9.2.3	ANAC_PDRCON_PDRCFG_Get.....	111
9.2.4	ANAC_PDRCON_PDREN_Setable.....	111
9.2.5	ANAC_PDRCON_PDREN_Getable.....	111
9.2.6	ANAC_BORCON_BOR_PDRCFG_Set.....	112
9.2.7	ANAC_BORCON_BOR_PDRCFG_Get.....	112
9.2.8	ANAC_BORCON_OFF_BOR_Setable.....	112
9.2.9	ANAC_BORCON_OFF_BOR_Getable.....	113
9.2.10	ANAC_SVD_Init.....	113
9.2.11	ANAC_SVDCON_SVDEN_Setable.....	114
9.2.12	ANAC_SVDCON_SVDEN_Getable.....	114
9.2.13	ANAC_SVDCFG_SVDMOD_Set.....	115
9.2.14	ANAC_SVDCFG_SVDMOD_Get.....	115
9.2.15	ANAC_SVDCFG_PFIE_Setable.....	115
9.2.16	ANAC_SVDCFG_PFIE_Getable.....	115
9.2.17	ANAC_SVDCFG_PRIIE_Setable.....	116
9.2.18	ANAC_SVDCFG_PRIIE_Getable.....	116
9.2.19	ANAC_SVDCFG_SVDLVL_Set.....	116
9.2.20	ANAC_SVDCFG_SVDLVL_Get.....	117
9.2.21	ANAC_SVDCFG_DFEN_Setable.....	118
9.2.22	ANAC_SVDCFG_DFEN_Getable.....	118
9.2.23	ANAC_SVDCFG_SVDITVL_Set.....	118
9.2.24	ANAC_SVDCFG_SVDITVL_Get.....	119
9.2.25	ANAC_SVDCON_SVDTE_Setable.....	119
9.2.26	ANAC_SVDCON_SVDTE_Getable.....	119
9.2.27	ANAC_SVDSIF_SVDO_Chk.....	119
9.2.28	ANAC_SVDSIF_PFF_Clr.....	120
9.2.29	ANAC_SVDSIF_PFF_Chk.....	120
9.2.30	ANAC_SVDSIF_PRF_Clr.....	120
9.2.31	ANAC_SVDSIF_PRF_Chk.....	121
9.2.32	ANAC_SVDALM_SVDALM_B_Chk.....	121
9.2.33	ANAC_FDETIF_FDETO_Chk.....	121
9.2.34	ANAC_FDETIE_FDET_IE_Setable.....	121
9.2.35	ANAC_FDETIE_FDET_IE_Getable.....	122
9.2.36	ANAC_FDETIF_FDETIF_Clr.....	122
9.2.37	ANAC_FDETIF_FDETIF_Chk.....	122
9.2.38	ANAC_ADC_Init.....	123



9.2.39	ANAC_ADCCON_ADC_EN_Setable.....	123
9.2.40	ANAC_ADCCON_ADC_EN_Getable.....	124
9.2.41	ANAC_ADCCON_ADC_IE_Setable.....	124
9.2.42	ANAC_ADCCON_ADC_IE_Getable.....	124
9.2.43	ANAC_ADCCON_ADC_VANA_EN_Set.....	125
9.2.44	ANAC_ADCCON_ADC_VANA_EN_Get.....	125
9.2.45	ANAC_ADCINSEL_BUFSEL_Set.....	125
9.2.46	ANAC_ADCINSEL_BUFSEL_Get.....	126
9.2.47	ANAC_ADCTRIM_Write.....	126
9.2.48	ANAC_ADCTRIM_Read.....	127
9.2.49	ANAC_ADCDATA_Read.....	127
9.2.50	ANAC_ADCIF_ADC_IF_Clr.....	127
9.2.51	ANAC_ADCIF_ADC_IF_Chk.....	127
9.2.52	ANAC_ADCINSEL_BUFEN_Setable.....	128
9.2.53	ANAC_ADCINSEL_BUFEN_Getable.....	128
9.2.54	ANAC_ADCINSEL_BUFBYP_Setable.....	128
9.2.55	ANAC_ADCINSEL_BUFBYP_Getable.....	128
9.2.56	ANAC_ADC_Channel_SetEx.....	129
9.2.57	ANAC_ADC_VoltageCalc.....	129
9.2.58	ANAC_ADC_TemperatureCalc.....	130
9.2.59	ANAC_COMPx_Init.....	130
9.2.60	ANAC_COMP1CR_CMP1O_Chk.....	131
9.2.61	ANAC_COMP1CR_V1PSEL_Set.....	131
9.2.62	ANAC_COMP1CR_V1PSEL_Get.....	131
9.2.63	ANAC_COMP1CR_V1NSEL_Set.....	132
9.2.64	ANAC_COMP1CR_V1NSEL_Get.....	132
9.2.65	ANAC_COMP1CR_CMP1EN_Setable.....	132
9.2.66	ANAC_COMP1CR_CMP1EN_Getable.....	133
9.2.67	ANAC_COMP1CR_CMP1DF_Setable.....	133
9.2.68	ANAC_COMP1CR_CMP1DF_Getable.....	133
9.2.69	ANAC_COMP1CR_CMP1SEL_Set.....	134
9.2.70	ANAC_COMP1CR_CMP1SEL_Get.....	134
9.2.71	ANAC_COMP1CR_CMP1IE_Setable.....	134
9.2.72	ANAC_COMP1CR_CMP1IE_Getable.....	135
9.2.73	ANAC_COMP1IF_CMP1IF_Clr.....	135
9.2.74	ANAC_COMP1IF_CMP1IF_Chk.....	135
9.2.75	ANAC_COMP2CR_CMP2O_Chk.....	135
9.2.76	ANAC_COMP2CR_V2PSEL_Set.....	136
9.2.77	ANAC_COMP2CR_V2PSEL_Get.....	136
9.2.78	ANAC_COMP2CR_V2NSEL_Set.....	136



9.2.79	<i>ANAC_COMP2CR_V2NSEL_Get</i> .....	137
9.2.80	<i>ANAC_COMP2CR_CMP2EN_Setable</i> .....	137
9.2.81	<i>ANAC_COMP2CR_CMP2EN_Getable</i> .....	137
9.2.82	<i>ANAC_COMPICR_CMP2DF_Setable</i> .....	138
9.2.83	<i>ANAC_COMPICR_CMP2DF_Getable</i> .....	138
9.2.84	<i>ANAC_COMPICR_CMP2SEL_Set</i> .....	138
9.2.85	<i>ANAC_COMPICR_CMP2SEL_Get</i> .....	138
9.2.86	<i>ANAC_COMPICR_CMP2IE_Setable</i> .....	139
9.2.87	<i>ANAC_COMPICR_CMP2IE_Getable</i> .....	139
9.2.88	<i>ANAC_COMPIF_CMP2IF_Clr</i> .....	139
9.2.89	<i>ANAC_COMPIF_CMP2IF_Chk</i> .....	140
9.2.90	<i>ANAC_COMPICR_BUFBY_P_Setable</i> .....	140
9.2.91	<i>ANAC_COMPICR_BUFBY_P_Getable</i> .....	140
9.2.92	<i>ANAC_COMPICR_BUFENB_Setable</i> .....	140
9.2.93	<i>ANAC_COMPICR_BUFENB_Getable</i> .....	141
<b>10</b>	<b>BTIM</b> .....	<b>141</b>
10.1	BTIM 寄存器结构.....	141
10.2	BTIM 库函数 .....	142
10.2.1	<i>BTIMx_Init</i> .....	145
10.2.2	<i>BTIMx_Deinit</i> .....	148
10.2.3	<i>BTIMx_BTCR1_CHEN_Setable</i> .....	148
10.2.4	<i>BTIMx_BTCR1_CHEN_Getable</i> .....	148
10.2.5	<i>BTIMx_BTCR1_CLEN_Setable</i> .....	148
10.2.6	<i>BTIMx_BTCR1_CLEN_Getable</i> .....	149
10.2.7	<i>BTIMx_BTCR1_MODE_Set</i> .....	149
10.2.8	<i>BTIMx_BTCR1_MODE_Get</i> .....	149
10.2.9	<i>BTIMx_BTCR1_EDGESEL_Set</i> .....	150
10.2.10	<i>BTIMx_BTCR1_EDGESEL_Get</i> .....	150
10.2.11	<i>BTIMx_BTCR1_CAPMOD_Set</i> .....	150
10.2.12	<i>BTIMx_BTCR1_CAPMOD_Get</i> .....	151
10.2.13	<i>BTIMx_BTCR1_CAPCLR_Set</i> .....	151
10.2.14	<i>BTIMx_BTCR1_CAPCLR_Get</i> .....	151
10.2.15	<i>BTIMx_BTCR1_CAPONCE_Set</i> .....	152
10.2.16	<i>BTIMx_BTCR1_CAPONCE_Get</i> .....	152
10.2.17	<i>BTIMx_BTCR1_PWM_Setable</i> .....	152
10.2.18	<i>BTIMx_BTCR1_PWM_Getable</i> .....	153
10.2.19	<i>BTIMx_BTCR2_SIG2SEL_Set</i> .....	153
10.2.20	<i>BTIMx_BTCR2_SIG2SEL_Get</i> .....	153
10.2.21	<i>BTIMx_BTCR2_SIG1SEL_Set</i> .....	153



10.2.22	<i>BTIMx_BTCR2_SIG1SEL_Get</i> .....	154
10.2.23	<i>BTIMx_BTCR2_CNTHSEL_Set</i> .....	154
10.2.24	<i>BTIMx_BTCR2_CNTHSEL_Get</i> .....	154
10.2.25	<i>BTIMx_BTCR2_DIREN_Setable</i> .....	155
10.2.26	<i>BTIMx_BTCR2_DIREN_Getable</i> .....	155
10.2.27	<i>BTIMx_BTCR2_STDIR_Setable</i> .....	155
10.2.28	<i>BTIMx_BTCR2_STDIR_Getable</i> .....	156
10.2.29	<i>BTIMx_BTCR2_SRCSEL_Set</i> .....	156
10.2.30	<i>BTIMx_BTCR2_SRCSEL_Get</i> .....	156
10.2.31	<i>BTIMx_BTCFG1_RTCSEL1_Set</i> .....	157
10.2.32	<i>BTIMx_BTCFG1_RTCSEL1_Get</i> .....	157
10.2.33	<i>BTIMx_BTCFG1_GRP1SEL_Set</i> .....	157
10.2.34	<i>BTIMx_BTCFG1_GRP1SEL_Get</i> .....	158
10.2.35	<i>BTIMx_BTCFG2_INSEL1_Get</i> .....	158
10.2.36	<i>BTIMx_BTCFG2_INSEL1_Get</i> .....	158
10.2.37	<i>BTIMx_BTCFG2_EXSEL1_Set</i> .....	159
10.2.38	<i>BTIMx_BTCFG2_EXSEL1_Get</i> .....	159
10.2.39	<i>BTIMx_BTCFG1_RTCSEL2_Set</i> .....	159
10.2.40	<i>BTIMx_BTCFG1_RTCSEL2_Get</i> .....	160
10.2.41	<i>BTIMx_BTCFG1_GRP2SEL_Set</i> .....	160
10.2.42	<i>BTIMx_BTCFG1_GRP2SEL_Get</i> .....	160
10.2.43	<i>BTIMx_BTCFG2_EXSEL2_Set</i> .....	161
10.2.44	<i>BTIMx_BTCFG2_EXSEL2_Get</i> .....	161
10.2.45	<i>BTIMx_BTCFG2_INSEL2_Set</i> .....	162
10.2.46	<i>BTIMx_BTCFG2_INSEL2_Get</i> .....	162
10.2.47	<i>BTIMx_BTCR2_DIRPO_Set</i> .....	162
10.2.48	<i>BTIMx_BTCR2_DIRPO_Get</i> .....	163
10.2.49	<i>BTIMx_BTPRES_Write</i> .....	163
10.2.50	<i>BTIMx_BTPRES_Read</i> .....	163
10.2.51	<i>BTIMx_BTLOADCR_LHEN_Setable</i> .....	163
10.2.52	<i>BTIMx_BTLOADCR_LLEN_Setable</i> .....	164
10.2.53	<i>BTIMx_BTCNTL_Read</i> .....	164
10.2.54	<i>BTIMx_BTCNTH_Read</i> .....	164
10.2.55	<i>BTIMx_BTPRESET_PRESETH_Set</i> .....	165
10.2.56	<i>BTIMx_BTPRESET_PRESETH_Get</i> .....	165
10.2.57	<i>BTIMx_BTPRESET_PRESETL_Set</i> .....	165
10.2.58	<i>BTIMx_BTPRESET_PRESETL_Get</i> .....	165
10.2.59	<i>BTIMx_BTLOADL_Write</i> .....	166
10.2.60	<i>BTIMx_BTLOADL_Read</i> .....	166
10.2.61	<i>BTIMx_BTLOADH_Write</i> .....	166



10.2.62	<i>BTIMx_BTLOADH_Read</i> .....	166
10.2.63	<i>BTIMx_BTCMPL_Write</i> .....	167
10.2.64	<i>BTIMx_BTCMPL_Read</i> .....	167
10.2.65	<i>BTIMx_BTCMPH_Write</i> .....	167
10.2.66	<i>BTIMx_BTCMPH_Read</i> .....	167
10.2.67	<i>BTIMx_BTOUTCNT_Write</i> .....	168
10.2.68	<i>BTIMx_BTOUTCNT_Read</i> .....	168
10.2.69	<i>BTIMx_BTOCR_OUTCLR_Set</i> .....	168
10.2.70	<i>BTIMx_BTOCR_OUTCLR_Get</i> .....	168
10.2.71	<i>BTIMx_BTOCR_OUTINV_Set</i> .....	169
10.2.72	<i>BTIMx_BTOCR_OUTINV_Get</i> .....	169
10.2.73	<i>BTIMx_BTOCR_OUTMOD_Set</i> .....	169
10.2.74	<i>BTIMx_BTOCR_OUTMOD_Get</i> .....	170
10.2.75	<i>BTIMx_BTOCR_OUTSEL_Set</i> .....	170
10.2.76	<i>BTIMx_BTOCR_OUTSEL_Get</i> .....	170
10.2.77	<i>BTIMx_BTIE_CMPHIE_Setable</i> .....	171
10.2.78	<i>BTIMx_BTIE_CMPHIE_Getable</i> .....	171
10.2.79	<i>BTIMx_BTIE_CMPLIE_Setable</i> .....	171
10.2.80	<i>BTIMx_BTIE_CMPLIE_Getable</i> .....	172
10.2.81	<i>BTIMx_BTIE_OVHIE_Setable</i> .....	172
10.2.82	<i>BTIMx_BTIE_OVHIE_Getable</i> .....	172
10.2.83	<i>BTIMx_BTIE_OVLIE_Setable</i> .....	172
10.2.84	<i>BTIMx_BTIE_OVLIE_Getable</i> .....	173
10.2.85	<i>BTIMx_BTIE_CAPIE_Setable</i> .....	173
10.2.86	<i>BTIMx_BTIE_CAPIE_Getable</i> .....	173
10.2.87	<i>BTIMx_BTIF_EDGESTA_Clr</i> .....	174
10.2.88	<i>BTIMx_BTIF_EDGESTA_Chk</i> .....	174
10.2.89	<i>BTIMx_BTIF_CMPHIF_Clr</i> .....	174
10.2.90	<i>BTIMx_BTIF_CMPHIF_Chk</i> .....	174
10.2.91	<i>BTIMx_BTIF_CMPLIF_Clr</i> .....	175
10.2.92	<i>BTIMx_BTIF_CMPLIF_Chk</i> .....	175
10.2.93	<i>BTIMx_BTIF_OVHIF_Clr</i> .....	175
10.2.94	<i>BTIMx_BTIF_OVHIF_Chk</i> .....	175
10.2.95	<i>BTIMx_BTIF_OVLIF_Clr</i> .....	176
10.2.96	<i>BTIMx_BTIF_OVLIF_Chk</i> .....	176
10.2.97	<i>BTIMx_BTIF_CAPIF_Clr</i> .....	176
10.2.98	<i>BTIMx_BTIF_CAPIF_Chk</i> .....	177
<b>11</b>	<b>LPTIM</b> .....	<b>178</b>
11.1	LPTIM 寄存器结构 .....	178



11.2	定时器中断库函数.....	178
11.2.1	LPTIM_Init.....	179
11.2.2	LPTIM_Deinit.....	180
11.2.3	LPTIM_LPTCTRL_LPTEN_Setable.....	181
11.2.4	LPTIM_LPTCTRL_LPTEN_Getable.....	181
11.2.5	LPTIM_LPTCFG_CLKSEL_Set.....	181
11.2.6	LPTIM_LPTCFG_CLKSEL_Get.....	182
11.2.7	LPTIM_LPTCFG_DIVSEL_Set.....	182
11.2.8	LPTIM_LPTCFG_DIVSEL_Get.....	182
11.2.9	LPTIM_LPTCFG_TMODE_Set.....	183
11.2.10	LPTIM_LPTCFG_TMODE_Get.....	183
11.2.11	LPTIM_LPTCNT_Write.....	183
11.2.12	LPTIM_LPTCNT_Read.....	184
11.2.13	LPTIM_LPTCMP_Write.....	184
11.2.14	LPTIM_LPTCMP_Read.....	184
11.2.15	LPTIM_LPTTARGET_Write.....	184
11.2.16	LPTIM_LPTTARGET_Read.....	185
11.2.17	LPTIM_LPTCFG_MODE_Set.....	185
11.2.18	LPTIM_LPTCFG_MODE_Get.....	185
11.2.19	LPTIM_LPTCFG_EDGESEL_Set.....	185
11.2.20	LPTIM_LPTCFG_EDGESEL_Get.....	186
11.2.21	LPTIM_LPTCFG_TRIGCFG_Set.....	186
11.2.22	LPTIM_LPTCFG_TRIGCFG_Get.....	186
11.2.23	LPTIM_LPTCFG_POLARITY_Set.....	187
11.2.24	LPTIM_LPTCFG_POLARITY_Get.....	187
11.2.25	LPTIM_LPTIE_OVIE_Setable.....	187
11.2.26	LPTIM_LPTIE_OVIE_Getable.....	188
11.2.27	LPTIM_LPTIE_COMPIE_Setable.....	188
11.2.28	LPTIM_LPTIE_COMPIE_Getable.....	188
11.2.29	LPTIM_LPTIE_TRIGIE_Setable.....	188
11.2.30	LPTIM_LPTIE_TRIGIE_Getable.....	189
11.2.31	LPTIM_LPTIF_OVIF_Clr.....	189
11.2.32	LPTIM_LPTIF_OVIF_Chk.....	189
11.2.33	LPTIM_LPTIF_COMPIF_Clr.....	190
11.2.34	LPTIM_LPTIF_COMPIF_Chk.....	190
11.2.35	LPTIM_LPTIF_TRIGIF_Clr.....	190
11.2.36	LPTIM_LPTIF_TRIGIF_Chk.....	190
11.2.37	LPTIM_LPTCFG_FLTEN_Setable.....	191
11.2.38	LPTIM_LPTCFG_FLTEN_Getable.....	191
11.2.39	LPTIM_LPTCFG_PWM_Set.....	191



11.2.40	<i>LPTIM_LPTCFG_PWM_Get</i> .....	191
<b>12</b>	<b>ETIM</b> .....	<b>193</b>
12.1	ETIM 寄存器结构.....	193
12.2	ETIM 库函数.....	194
12.2.1	<i>ETIMx_Init</i> .....	195
12.2.2	<i>ETIMx_Deinit</i> .....	198
12.2.3	<i>ETIMx_ETxCR_CEN_Setable</i> .....	198
12.2.4	<i>ETIMx_ETxCR_CEN_Getable</i> .....	198
12.2.5	<i>ETIMx_ETxCR_MOD_Set</i> .....	199
12.2.6	<i>ETIMx_ETxCR_MOD_Get</i> .....	199
12.2.7	<i>ETIMx_ETxCR_EDGESEL_Set</i> .....	199
12.2.8	<i>ETIMx_ETxCR_EDGESEL_Get</i> .....	200
12.2.9	<i>ETIMx_ETxPESCALE1_Write</i> .....	200
12.2.10	<i>ETIMx_ETxPESCALE1_Read</i> .....	200
12.2.11	<i>ETIMx_ETxPESCALE2_Write</i> .....	200
12.2.12	<i>ETIMx_ETxPESCALE2_Read</i> .....	201
12.2.13	<i>ETIMx_ETxIVR_Write</i> .....	201
12.2.14	<i>ETIMx_ETxIVR_Read</i> .....	201
12.2.15	<i>ETIMx_ETxCMP_Write</i> .....	201
12.2.16	<i>ETIMx_ETxCMP_Read</i> .....	202
12.2.17	<i>ETIMx_ETxCR_PWM_Setable</i> .....	202
12.2.18	<i>ETIMx_ETxCR_PWM_Getable</i> .....	202
12.2.19	<i>ETIMx_ETxINSEL_SIG1SEL_Set</i> .....	203
12.2.20	<i>ETIMx_ETxINSEL_SIG1SEL_Get</i> .....	203
12.2.21	<i>ETIMx_ETxINSEL_SIG2SEL_Set</i> .....	203
12.2.22	<i>ETIMx_ETxINSEL_SIG2SEL_Get</i> .....	203
12.2.23	<i>ETIMx_ETxCR_CAPEDGE_Set</i> .....	204
12.2.24	<i>ETIMx_ETxCR_CAPEDGE_Get</i> .....	204
12.2.25	<i>ETIMx_ETxCR_CAPMOD_Set</i> .....	204
12.2.26	<i>ETIMx_ETxCR_CAPMOD_Get</i> .....	205
12.2.27	<i>ETIMx_ETxCR_CAPONCE_Setable</i> .....	205
12.2.28	<i>ETIMx_ETxCR_CAPONCE_Getable</i> .....	205
12.2.29	<i>ETIMx_ETxCR_CAPCLR_Setable</i> .....	206
12.2.30	<i>ETIMx_ETxCR_CAPCLR_Getable</i> .....	206
12.2.31	<i>ETIMx_ETxINSEL_GRP1SEL_Set</i> .....	206
12.2.32	<i>ETIMx_ETxINSEL_GRP1SEL_Get</i> .....	207
12.2.33	<i>ETIMx_ETxINSEL_GRP2SEL_Set</i> .....	208
12.2.34	<i>ETIMx_ETxINSEL_GRP2SEL_Get</i> .....	209
12.2.35	<i>ETIMx_ETxCR_CASEN_Setable</i> .....	210





12.2.36	<i>ETIMx_ETxCR_CASEN_Getable</i> .....	210
12.2.37	<i>ETIMx_ETxIE_CMPIE_Setable</i> .....	210
12.2.38	<i>ETIMx_ETxIE_CMPIE_Getable</i> .....	211
12.2.39	<i>ETIMx_ETxIE_CAPIE_Setable</i> .....	211
12.2.40	<i>ETIMx_ETxIE_CAPIE_Getable</i> .....	211
12.2.41	<i>ETIMx_ETxIE_OVIE_Setable</i> .....	211
12.2.42	<i>ETIMx_ETxIE_OVIE_Getable</i> .....	212
12.2.43	<i>ETIMx_ETxIF_CMPIF_Clr</i> .....	212
12.2.44	<i>ETIMx_ETxIF_CMPIF_Chk</i> .....	212
12.2.45	<i>ETIMx_ETxIF_CAPIF_Clr</i> .....	213
12.2.46	<i>ETIMx_ETxIF_CAPIF_Chk</i> .....	213
12.2.47	<i>ETIMx_ETxIF_OVIF_Clr</i> .....	213
12.2.48	<i>ETIMx_ETxIF_OVIF_Chk</i> .....	213
12.2.49	<i>ETIMx_ETxIF_EDGESTA_Chk</i> .....	214
12.2.50	<i>ETIMx_ETxCR_EXFLT_Setable</i> .....	214
12.2.51	<i>ETIMx_ETxCR_EXFLT_Getable</i> .....	214
<b>13</b>	<b>RTC</b> .....	<b>215</b>
13.1	RTC 结构.....	215
13.2	RTC 库函数.....	215
13.2.1	<i>RTC_Deinit</i> .....	217
13.2.2	<i>RTC_RTCWE_Write</i> .....	218
13.2.3	<i>RTC_BCDSEC_Write</i> .....	218
13.2.4	<i>RTC_BCDSEC_Read</i> .....	218
13.2.5	<i>RTC_BCDMIN_Write</i> .....	218
13.2.6	<i>RTC_BCDMIN_Read</i> .....	219
13.2.7	<i>RTC_BCDHOUR_Write</i> .....	219
13.2.8	<i>RTC_BCDHOUR_Read</i> .....	219
13.2.9	<i>RTC_BCDDATE_Write</i> .....	219
13.2.10	<i>RTC_BCDDATE_Read</i> .....	220
13.2.11	<i>RTC_BCDWEEK_Write</i> .....	220
13.2.12	<i>RTC_BCDWEEK_Read</i> .....	220
13.2.13	<i>RTC_BCDMONTH_Write</i> .....	220
13.2.14	<i>RTC_BCDMONTH_Read</i> .....	221
13.2.15	<i>RTC_BCDYEAR_Write</i> .....	221
13.2.16	<i>RTC_BCDYEAR_Read</i> .....	221
13.2.17	<i>RTC_MSECCNT_Write</i> .....	221
13.2.18	<i>RTC_MSECCNT_Read</i> .....	222
13.2.19	<i>RTC_ALARM_ALARMHOUR_Set</i> .....	222
13.2.20	<i>RTC_ALARM_ALARMHOUR_Get</i> .....	222





13.2.21	<i>RTC_ALARM_ALARMMIN_Set</i> .....	222
13.2.22	<i>RTC_ALARM_ALARMMIN_Get</i> .....	223
13.2.23	<i>RTC_ALARM_ALARMSEC_Set</i> .....	223
13.2.24	<i>RTC_ALARM_ALARMSEC_Get</i> .....	223
13.2.25	<i>RTC_FSEL_FSEL_Set</i> .....	223
13.2.26	<i>RTC_FSEL_FSEL_Get</i> .....	224
13.2.27	<i>RTC_STAMPEN_STAMP0EN_Setable</i> .....	225
13.2.28	<i>RTC_STAMPEN_STAMP0EN_Getable</i> .....	225
13.2.29	<i>RTC_STAMPEN_STAMP1EN_Setable</i> .....	225
13.2.30	<i>RTC_STAMPEN_STAMP1EN_Getable</i> .....	225
13.2.31	<i>RTC_CLKSTAMP0R_Read</i> .....	226
13.2.32	<i>RTC_CALSTAMP0R_Read</i> .....	226
13.2.33	<i>RTC_CLKSTAMP0F_Read</i> .....	226
13.2.34	<i>RTC_CALSTAMP0F_Read</i> .....	227
13.2.35	<i>RTC_CLKSTAMP1R_Read</i> .....	227
13.2.36	<i>RTC_CALSTAMP1R_Read</i> .....	227
13.2.37	<i>RTC_CLKSTAMP1F_Read</i> .....	228
13.2.38	<i>RTC_CALSTAMP1F_Read</i> .....	228
13.2.39	<i>RTC_RTCIE_SetableEx</i> .....	228
13.2.40	<i>RTC_RTCIE_GetableEx</i> .....	229
13.2.41	<i>RTC_RTCIF_ClrEx</i> .....	229
13.2.42	<i>RTC_RTCIF_ChkEx</i> .....	229
13.2.43	<i>RTC_TimeDate_SetEx</i> .....	229
13.2.44	<i>RTC_TimeDate_GetEx</i> .....	230
13.2.45	<i>RTC_AlarmTime_SetEx</i> .....	230
13.2.46	<i>RTC_AlarmTime_GetEx</i> .....	231
13.2.47	<i>RTC_CLKSTAMPxx_GetEx</i> .....	231
13.2.48	<i>RTC_ADJUST_Write</i> .....	231
13.2.49	<i>RTC_ADJUST_Read</i> .....	232
13.2.50	<i>RTC_ADSIGN_ADSIGN_Set</i> .....	232
13.2.51	<i>RTC_ADSIGN_ADSIGN_Get</i> .....	232
13.2.52	<i>RTC_PR1SEN_PR1SEN_Setable</i> .....	232
13.2.53	<i>RTC_PR1SEN_PR1SEN_Getable</i> .....	233
13.2.54	<i>RTC_Trim_Proc</i> .....	233
<b>14</b>	<b>AES</b> .....	<b>234</b>
14.1	AES 寄存器结构.....	234
14.2	AES 库函数.....	234
14.2.1	<i>AES_Init</i> .....	235
14.2.2	<i>AES_Deinit</i> .....	236



14.2.3	<i>AES_AESCR_AESEN_Setable</i> .....	237
14.2.4	<i>AES_AESCR_AESEN_Getable</i> .....	237
14.2.5	<i>AES_AESCR_MODE_Set</i> .....	237
14.2.6	<i>AES_AESCR_MODE_Get</i> .....	238
14.2.7	<i>AES_AESCR_CHMOD_Set</i> .....	238
14.2.8	<i>AES_AESCR_CHMOD_Get</i> .....	238
14.2.9	<i>AES_AESCR_DATATYP_Set</i> .....	239
14.2.10	<i>AES_AESCR_DATATYP_Get</i> .....	239
14.2.11	<i>AES_AESDIN_Write</i> .....	239
14.2.12	<i>AES_AESDIN_Read</i> .....	240
14.2.13	<i>AES_AESDOUT_Write</i> .....	240
14.2.14	<i>AES_AESDOUT_Read</i> .....	240
14.2.15	<i>AES_AESCR_KEYLEN_Set</i> .....	240
14.2.16	<i>AES_AESCR_KEYLEN_Get</i> .....	241
14.2.17	<i>AES_AESDIN_GroupWrite_128BIT</i> .....	241
14.2.18	<i>AES_AESDOUT_GroupRead_128BIT</i> .....	241
14.2.19	<i>AES_GroupWriteAndRead_128BIT</i> .....	242
14.2.20	<i>AES_AESKEY_WriteEx</i> .....	242
14.2.21	<i>AES_AESKEY_ReadEx</i> .....	242
14.2.22	<i>AES_AESIVR_WriteEx</i> .....	242
14.2.23	<i>AES_AESIVR_ReadEx</i> .....	243
14.2.24	<i>AES_AESCR_ERRIE_Setable</i> .....	243
14.2.25	<i>AES_AESCR_ERRIE_Getable</i> .....	243
14.2.26	<i>AES_AESCR_CCFIE_Setable</i> .....	243
14.2.27	<i>AES_AESCR_CCFIE_Getable</i> .....	244
14.2.28	<i>AES_AESCR_ERRC_Clr</i> .....	244
14.2.29	<i>AES_AESCR_CCFC_Clr</i> .....	244
14.2.30	<i>AES_AESIF_WRERR_Chk</i> .....	244
14.2.31	<i>AES_AESIF_RDERR_Chk</i> .....	245
14.2.32	<i>AES_AESIF_CCF_Chk</i> .....	245
14.2.33	<i>AES_AESCR_DMAIEN_Setable</i> .....	245
14.2.34	<i>AES_AESCR_DMAIEN_Getable</i> .....	246
14.2.35	<i>AES_AESCR_DMAOEN_Setable</i> .....	246
14.2.36	<i>AES_AESCR_DMAOEN_Getable</i> .....	246
<b>15</b>	<b>CRC</b> .....	<b>247</b>
15.1	CRC 寄存器结构 .....	247
15.2	CRC 库函数 .....	247
15.2.1	<i>CRC_Init</i> .....	248
15.2.2	<i>CRC_Deinit</i> .....	248



15.2.3	<i>CRC_CRCDDR_Write</i> .....	248
15.2.4	<i>CRC_CRCDDR_Read</i> .....	249
15.2.5	<i>CRC_CRCCR_RFLTIN_Setable</i> .....	249
15.2.6	<i>CRC_CRCCR_RFLTIN_Getable</i> .....	249
15.2.7	<i>CRC_CRCCR_RFLT0_Setable</i> .....	250
15.2.8	<i>CRC_CRCCR_RFLT0_Getable</i> .....	250
15.2.9	<i>CRC_CRCCR_RES_Chk</i> .....	250
15.2.10	<i>CRC_CRCCR_BUSY_Chk</i> .....	250
15.2.11	<i>CRC_CRCCR_XOR_Setable</i> .....	251
15.2.12	<i>CRC_CRCCR_XOR_Getable</i> .....	251
15.2.13	<i>CRC_CRCCR_CRCSEL_Set</i> .....	251
15.2.14	<i>CRC_CRCCR_CRCSEL_Get</i> .....	252
15.2.15	<i>CRC_CRCLFSR_Write</i> .....	252
15.2.16	<i>CRC_CRCLFSR_Read</i> .....	252
15.2.17	<i>CRC_CRCXOR_Write</i> .....	252
15.2.18	<i>CRC_CRCXOR_Read</i> .....	253
15.2.19	<i>CRC_CRCFLSEN_FLSCRCEN_Setable</i> .....	253
15.2.20	<i>CRC_CRCFLSEN_FLSCRCEN_Getable</i> .....	253
15.2.21	<i>CRC_CRCFLSAD_Write</i> .....	253
15.2.22	<i>CRC_CRCFLSAD_Read</i> .....	254
15.2.23	<i>CRC_CRCFLSSIZE_Write</i> .....	254
15.2.24	<i>CRC_CRCFLSSIZE_Read</i> .....	254
<b>16</b>	<b>DMA</b> .....	<b>255</b>
16.1	DMA 寄存器结构 .....	255
16.2	DMA 库函数 .....	255
16.2.1	<i>DMA_Init</i> .....	256
16.2.2	<i>DMA_Deinit</i> .....	257
16.2.3	<i>DMA_GLOBALCTRL_DMAEN_Setable</i> .....	258
16.2.4	<i>DMA_GLOBALCTRL_DMAEN_Getable</i> .....	258
16.2.5	<i>DMA_CHxCTRL_EN_Setable</i> .....	258
16.2.6	<i>DMA_CHxCTRL_TSIZE_Set</i> .....	259
16.2.7	<i>DMA_CHxCTRL_TSIZE_Get</i> .....	259
16.2.8	<i>DMA_CHxCTRL_PRI_Set</i> .....	259
16.2.9	<i>DMA_CHxCTRL_PRI_Get</i> .....	259
16.2.10	<i>DMA_CHxCTRL_INC_Set</i> .....	260
16.2.11	<i>DMA_CHxCTRL_INC_Get</i> .....	260
16.2.12	<i>DMA_CHxCTRL_SSEL_Set</i> .....	260
16.2.13	<i>DMA_CHxCTRL_SSEL_Get</i> .....	261
16.2.14	<i>DMA_CHxCTRL_FTIE_Setable</i> .....	261



16.2.15	<i>DMA_CHxCTRL_FTIE_Getable</i> .....	262
16.2.16	<i>DMA_CHxCTRL_HTIE_Setable</i> .....	262
16.2.17	<i>DMA_CHxCTRL_HTIE_Getable</i> .....	262
16.2.18	<i>DMA_CHSTATUS_DMACHxHT_Clr</i> .....	263
16.2.19	<i>DMA_CHSTATUS_DMACHxHT_Chk</i> .....	263
16.2.20	<i>DMA_CHSTATUS_DMACHxFT_Clr</i> .....	263
16.2.21	<i>DMA_CHSTATUS_DMACHxFT_Chk</i> .....	263
16.2.22	<i>DMA_CHxRAMAD_Write</i> .....	264
16.2.23	<i>DMA_CH7FLSAD_Write</i> .....	264
16.2.24	<i>DMA_CH7RAMAD_Write</i> .....	264
<b>17</b>	<b>TRNG</b> .....	<b>265</b>
17.1	TRNG 寄存器结构.....	265
17.2	TRNG 库函数.....	265
17.2.1	<i>TRNG_Deinit</i> .....	265
17.2.2	<i>TRNG_RNGOUT_Read</i> .....	266
17.2.3	<i>TRNG_CRCCON_CRCEN_Setable</i> .....	266
17.2.4	<i>TRNG_CRCCON_CRCEN_Getable</i> .....	266
17.2.5	<i>TRNG_RNGIE_RNGIE_Setable</i> .....	267
17.2.6	<i>TRNG_RNGIE_RNGIE_Getable</i> .....	267
17.2.7	<i>TRNG_RNGIF_LFSREN_Chk</i> .....	267
17.2.8	<i>TRNG_RNGIF_RNF_Clr</i> .....	267
17.2.9	<i>TRNG_RNGIF_RNF_Chk</i> .....	268
17.2.10	<i>TRNG_CRCIN_Write</i> .....	268
17.2.11	<i>TRNG_CRCIN_Read</i> .....	268
17.2.12	<i>TRNG_CRCFLAG_CRCD_Clr</i> .....	268
17.2.13	<i>TRNG_CRCFLAG_CRCD_Chk</i> .....	269
17.2.14	<i>ANAC_TRNGCON_TRNGEN_Setable</i> .....	269
17.2.15	<i>ANAC_TRNGCON_TRNGEN_Getable</i> .....	269
<b>18</b>	<b>U7816</b> .....	<b>270</b>
18.1	U7816 寄存器结构.....	270
18.2	U7816 库函数.....	270
18.2.1	<i>U7816_Init</i> .....	272
18.2.2	<i>U7816x_Deinit</i> .....	274
18.2.3	<i>U7816x_U7816CTRL_TXEN_Setable</i> .....	274
18.2.4	<i>U7816x_U7816CTRL_TXEN_Getable</i> .....	275
18.2.5	<i>U7816x_U7816CTRL_RXEN_Setable</i> .....	275
18.2.6	<i>U7816x_U7816CTRL_RXEN_Getable</i> .....	275
18.2.7	<i>U7816CTRL_CKOEN_Setable</i> .....	276



18.2.8	U7816CTRL_CKOEN_Getable.....	276
18.2.9	U7816x_U7816CTRL_HPUAT_Setable.....	276
18.2.10	U7816x_U7816CTRL_HPUAT_Getable.....	277
18.2.11	U7816x_U7816CTRL_HPUEN_Setable.....	277
18.2.12	U7816x_U7816CTRL_HPUEN_Getable.....	277
18.2.13	U7816x_U7816FRC_ERSW_Set.....	278
18.2.14	U7816x_U7816FRC_ERSW_Get.....	278
18.2.15	U7816x_U7816FRC_ERSGD_Set.....	278
18.2.16	U7816x_U7816FRC_ERSGD_Get.....	279
18.2.17	U7816x_U7816FRC_BG TEN_Setable.....	279
18.2.18	U7816x_U7816FRC_BG TEN_Getable.....	279
18.2.19	U7816x_U7816FRC_REP_T_Set.....	280
18.2.20	U7816x_U7816FRC_REP_T_Get.....	280
18.2.21	U7816x_U7816FRC_PAR_Set.....	280
18.2.22	U7816x_U7816FRC_PAR_Get.....	281
18.2.23	U7816x_U7816FRC_FREN_Setable.....	281
18.2.24	U7816x_U7816FRC_FREN_Getable.....	281
18.2.25	U7816x_U7816FRC_TREPEN_Setable.....	282
18.2.26	U7816x_U7816FRC_TREPEN_Getable.....	282
18.2.27	U7816x_U7816FRC_RREPEN_Setable.....	282
18.2.28	U7816x_U7816FRC_RREPEN_Getable.....	283
18.2.29	U7816x_U7816FRC_DICONV_Setable.....	283
18.2.30	U7816x_U7816FRC_DICONV_Getable.....	283
18.2.31	U7816x_U7816EGT_Write.....	284
18.2.32	U7816x_U7816EGT_Read.....	284
18.2.33	U7816x_U7816CLKDIV_Write.....	284
18.2.34	U7816x_U7816CLKDIV_Read.....	284
18.2.35	U7816x_U7816PDIV_Write.....	285
18.2.36	U7816x_U7816PDIV_Read.....	285
18.2.37	U7816x_U7816TXBUF_Write.....	285
18.2.38	U7816x_U7816RXBUF_Read.....	285
18.2.39	U7816x_U7816IE_RXIE_Setable.....	286
18.2.40	U7816x_U7816IE_RXIE_Getable.....	286
18.2.41	U7816x_U7816IE_TXIE_Setable.....	286
18.2.42	U7816x_U7816IE_TXIE_Getable.....	286
18.2.43	U7816x_U7816IE_LSIE_Setable.....	287
18.2.44	U7816x_U7816IE_LSIE_Getable.....	287
18.2.45	U7816x_U7816IF_ERRIF_Chk.....	287
18.2.46	U7816x_U7816IF_TXIF_Chk.....	288
18.2.47	U7816x_U7816IF_RXIF_Chk.....	288



18.2.48	U7816x_U7816ERR_TPARERR_Clr.....	288
18.2.49	U7816x_U7816ERR_TPARERR_Chk.....	288
18.2.50	U7816x_U7816ERR_RPARERR_Clr.....	289
18.2.51	U7816x_U7816ERR_RPARERR_Chk.....	289
18.2.52	U7816x_U7816ERR_FRERR_Clr.....	289
18.2.53	U7816x_U7816ERR_FRERR_Chk.....	290
18.2.54	U7816x_U7816ERR_OVERR_Clr.....	290
18.2.55	U7816x_U7816ERR_OVERR_Chk.....	290
18.2.56	U7816x_U7816STA_WAIT_RPT_Chk.....	290
18.2.57	U7816x_U7816STA_TXBUSY_Chk.....	291
18.2.58	U7816x_U7816STA_RXBUSY_Chk.....	291
<b>19</b>	<b>LCD.....</b>	<b>292</b>
19.1	LCD 寄存器结构.....	292
19.2	LCD 库函数 .....	293
19.2.1	LCD_Init.....	295
19.2.2	LCD_Deinit.....	298
19.2.3	LCD_DISPCTRL_LCDEN_Setable.....	298
19.2.4	LCD_DISPCTRL_LCDEN_Getable.....	298
19.2.5	LCD_DISPCTRL_FLICK_Setable.....	298
19.2.6	LCD_DISPCTRL_FLICK_Getable.....	299
19.2.7	LCD_DISPCTRL_TEST_Setable.....	299
19.2.8	LCD_DISPCTRL_TEST_Getable .....	299
19.2.9	LCD_LCDTEST_TESTEN_Setable.....	300
19.2.10	LCD_LCDTEST_TESTEN_Getable.....	300
19.2.11	LCD_DISPCTRL_DISPMD_Setable.....	300
19.2.12	LCD_DISPCTRL_DISPMD_Getable.....	300
19.2.13	LCD_LCDTEST_LCCTRL_Set.....	301
19.2.14	LCD_LCDTEST_LCCTRL_Get.....	301
19.2.15	LCD_DF_Write.....	301
19.2.16	LCD_DF_Read.....	302
19.2.17	LCD_TON_Write .....	302
19.2.18	LCD_TON_Read.....	302
19.2.19	LCD_TOFF_Write .....	302
19.2.20	LCD_TOFF_Read.....	303
19.2.21	LCD_DISPIE_DONIE_Setable.....	303
19.2.22	LCD_DISPIE_DONIE_Getable.....	303
19.2.23	LCD_DISPIE_DOFFIE_Setable.....	303
19.2.24	LCD_DISPIE_DOFFIE_Getable.....	304
19.2.25	LCD_DISPIF_DONIF_Clr.....	304



19.2.26	LCD_DISPIF_DONIF_Chk.....	304
19.2.27	LCD_DISPIF_DOFFIF_Clr.....	304
19.2.28	LCD_DISPIF_DOFFIF_Chk.....	305
19.2.29	LCD_LCDSET_BIASMD_Set.....	305
19.2.30	LCD_LCDSET_BIASMD_Get.....	305
19.2.31	LCD_LCDSET_WFT_Set.....	306
19.2.32	LCD_LCDSET_WFT_Get.....	306
19.2.33	LCD_LCDSET_LMUX_Set.....	306
19.2.34	LCD_LCDSET_LMUX_Get.....	306
19.2.35	LCD_LCDDRV_SCFSEL_Set.....	307
19.2.36	LCD_LCDDRV_SCFSEL_Get.....	307
19.2.37	LCD_LCDDRV_SC_CTRL_Set.....	308
19.2.38	LCD_LCDDRV_SC_CTRL_Get.....	308
19.2.39	LCD_LCDDRV_IC_CTRL_Set.....	308
19.2.40	LCD_LCDDRV_IC_CTRL_Get.....	309
19.2.41	LCD_LCDDRV_ENMODE_Set.....	309
19.2.42	LCD_LCDDRV_ENMODE_Get.....	309
19.2.43	LCD_LCDBIAS_Write.....	310
19.2.44	LCD_LCDBIAS_Read.....	310
19.2.45	LCD_LCDBSTCON_OSC_CFG_Set.....	311
19.2.46	LCD_LCDBSTCON_OSC_CFG_Get.....	311
19.2.47	LCD_LCDBSTCON_BUFENB_Setable.....	312
19.2.48	LCD_LCDBSTCON_BUFENB_Getable.....	312
19.2.49	LCD_LCDBSTCON_BUFBYP_Setable.....	312
19.2.50	LCD_LCDBSTCON_BUFBYP_Getable.....	313
19.2.51	LCD_LCDBSTCON_VLCDPD_Setable.....	313
19.2.52	LCD_LCDBSTCON_VLCDPD_Getable.....	313
19.2.53	LCD_LCDBSTCON_VLCDCFG_Set.....	314
19.2.54	LCD_LCDBSTCON_VLCDCFG_Get.....	314
19.2.55	LCD_LCDBSTCON_BOOST_TEN_Setable.....	314
19.2.56	LCD_LCDBSTCON_BOOST_TEN_Getable.....	315
19.2.57	LCD_LCDBSTCON_BOOST_EN_Setable.....	315
19.2.58	LCD_LCDBSTCON_BOOST_EN_Getable.....	315
19.2.59	LCD_COM_EN_Write.....	316
19.2.60	LCD_COM_EN_Read.....	316
19.2.61	LCD_SEG_EN0_Write.....	316
19.2.62	LCD_SEG_EN0_Read.....	317
19.2.63	LCD_SEG_EN1_Write.....	317
19.2.64	LCD_SEG_EN1_Read.....	317
19.2.65	LCD_DISPDATAx_Refresh.....	317





19.2.66	<i>LCD_Boost_Init</i> .....	318
19.2.67	<i>LCD_DISPCTRL_ANTIPOLAR_Setable</i> .....	319
19.2.68	<i>LCD_DISPCTRL_ANTIPOLAR_Getable</i> .....	319
<b>20</b>	<b>UART</b> .....	<b>319</b>
20.1	UART 寄存器结构.....	319
20.2	UART 库函数.....	321
20.2.1	<i>UART_Init</i> .....	323
20.2.2	<i>UART_SInit</i> .....	324
20.2.3	<i>UART_All_Deinit</i> .....	324
20.2.4	<i>UART_Deinit</i> .....	325
20.2.5	<i>UARTx_Deinit</i> .....	325
20.2.6	<i>UARTx_TXREG_Write</i> .....	325
20.2.7	<i>UARTx_RXREG_Read</i> .....	325
20.2.8	<i>UARTx_SPBRG_Write</i> .....	326
20.2.9	<i>UARTx_SPBRG_Read</i> .....	326
20.2.10	<i>UART_BaudREGCalc</i> .....	326
20.2.11	<i>UARTx_TXSTA_TXEN_Setable</i> .....	326
20.2.12	<i>UARTx_TXSTA_TXEN_Getable</i> .....	327
20.2.13	<i>UARTx_RXSTA_RXEN_Setable</i> .....	327
20.2.14	<i>UARTx_RXSTA_RXEN_Getable</i> .....	327
20.2.15	<i>UARTx_RXSTA_PDSEL_Set</i> .....	328
20.2.16	<i>UARTx_RXSTA_PDSEL_Get</i> .....	328
20.2.17	<i>UARTx_RTXCON_RTX7EN_Setable</i> .....	328
20.2.18	<i>UARTx_RTXCON_RTX7EN_Getable</i> .....	329
20.2.19	<i>UARTx_TXSTA_TX9D_Set</i> .....	329
20.2.20	<i>UARTx_RXSTA_RX9D_Chk</i> .....	329
20.2.21	<i>UARTx_TXSTA_STOPSEL_Set</i> .....	329
20.2.22	<i>UARTx_TXSTA_STOPSEL_Get</i> .....	330
20.2.23	<i>UARTx_TXSTA_TXIS_Set</i> .....	330
20.2.24	<i>UARTx_TXSTA_TXIS_Get</i> .....	330
20.2.25	<i>UARTx_TXBUFSTA_TX_INTSEL_Set</i> .....	331
20.2.26	<i>UARTx_TXBUFSTA_TX_INTSEL_Get</i> .....	331
20.2.27	<i>UARTx_TXBUFSTA_TXFF_Chk</i> .....	331
20.2.28	<i>UARTx_RXBUFSTA_RXFF_Chk</i> .....	332
20.2.29	<i>UART_UARTIE_RxTxIE_SetableEx</i> .....	332
20.2.30	<i>UART_UARTIE_RxTxIE_GetableEx</i> .....	332
20.2.31	<i>UART_UARTIF_RxTxIF_ClrEx</i> .....	333
20.2.32	<i>UART_UARTIF_RxTxIF_ChkEx</i> .....	333
20.2.33	<i>UARTx_RXSTA_ERRIE_Setable</i> .....	333





20.2.34	UARTx_RXSTA_ERRIE_Getable .....	334
20.2.35	UARTx_RXSTA_PERR_Clr.....	334
20.2.36	UARTx_RXSTA_PERR_Chk.....	334
20.2.37	UARTx_RXSTA_FERR_Clr.....	334
20.2.38	UARTx_RXSTA_FERR_Chk.....	335
20.2.39	UARTx_RXSTA_OERR_Clr.....	335
20.2.40	UARTx_RXSTA_OERR_Chk.....	335
20.2.41	UARTx_RTXCON_TXDFLAG_Setable.....	335
20.2.42	UARTx_RTXCON_TXDFLAG_Getable.....	336
20.2.43	UARTx_RTXCON_RXDFLAG_Setable.....	336
20.2.44	UARTx_RTXCON_RXDFLAG_Getable.....	336
20.2.45	UARTx_TXSTA_IREN_Setable.....	337
20.2.46	UARTx_TXSTA_IREN_Getable .....	337
20.2.47	UART_IRModulation_Init .....	337
20.2.48	UART_IRCON_TH_Set.....	338
20.2.49	UART_IRCON_TH_Get.....	338
20.2.50	UART_IRCON_TZBRG_Set.....	338
20.2.51	UART_IRCON_TZBRG_Get.....	338
20.2.52	UART_IRCON_IRFLAG_Setable.....	339
20.2.53	UART_IRCON_IRFLAG_Getable .....	339
<b>21</b>	<b>I2C .....</b>	<b>340</b>
21.1	I2C 寄存器结构 .....	340
21.2	I2C 库函数 .....	340
21.2.1	I2C_Deinit.....	341
21.2.2	I2C_I2CCTRL_I2CEN_Setable.....	342
21.2.3	I2C_I2CCTRL_I2CEN_Getable.....	342
21.2.4	I2C_I2CBUF_Write.....	342
21.2.5	I2C_I2CBUF_Read.....	343
21.2.6	I2C_I2CCTRL_SCLHL_Setable.....	343
21.2.7	I2C_I2CCTRL_SCLHL_Getable.....	343
21.2.8	I2C_I2CCTRL_SDAHL_Setable.....	343
21.2.9	I2C_I2CCTRL_SDAHL_Getable.....	344
21.2.10	I2C_I2CCTRL_ACKEN_Setable.....	344
21.2.11	I2C_I2CCTRL_ACKEN_Getable .....	344
21.2.12	I2C_I2CCTRL_RCEN_Setable.....	345
21.2.13	I2C_I2CCTRL_RCEN_Getable.....	345
21.2.14	I2C_I2CCTRL_RSEN_Setable.....	345
21.2.15	I2C_I2CCTRL_PEN_Setable .....	345
21.2.16	I2C_I2CCTRL_SEN_Setable.....	346



21.2.17	I2C_I2CBRG_Write .....	346
21.2.18	I2C_I2CBRG_Read .....	346
21.2.19	I2C_BaudREG_Calc .....	346
21.2.20	I2C_I2CSTA_RW_Chk .....	347
21.2.21	I2C_I2CSTA_BF_Chk .....	347
21.2.22	I2C_I2CSTA_S_Chk .....	347
21.2.23	I2C_I2CSTA_P_Chk .....	348
21.2.24	I2C_I2CSTA_WCOL_Clr .....	348
21.2.25	I2C_I2CSTA_WCOL_Chk .....	348
21.2.26	I2C_I2CSTA_ACKSTA_Clr .....	348
21.2.27	I2C_I2CSTA_ACKSTA_Chk .....	349
21.2.28	I2C_I2CSTA_ACKDT_Set .....	349
21.2.29	I2C_I2CSTA_ACKDT_Get .....	349
21.2.30	I2C_I2CIR_I2CIE_Setable .....	350
21.2.31	I2C_I2CIR_I2CIE_Getable .....	350
21.2.32	I2C_I2CIR_I2CIF_Clr .....	350
21.2.33	I2C_I2CIR_I2CIF_Chk .....	350
21.2.34	I2C_I2CFSM_I2CFSM_Get .....	351
21.2.35	I2C_I2CERR_ERRIE_Setable .....	351
21.2.36	I2C_I2CERR_ERRIE_Getable .....	352
21.2.37	I2C_I2CERR_OIERR_Clr .....	352
21.2.38	I2C_I2CERR_OIERR_Chk .....	352
21.2.39	I2C_I2CERR_SDERR_Clr .....	352
21.2.40	I2C_I2CERR_SDERR_Chk .....	353
21.2.41	I2C_I2CERR_IERR_Clr .....	353
21.2.42	I2C_I2CERR_IERR_Chk .....	353
<b>22</b>	<b>SPI .....</b>	<b>354</b>
22.1	SPI 寄存器结构 .....	354
22.2	SPI 库函数 .....	354
22.2.1	SPIx_Deinit .....	356
22.2.2	SPI_Master_Init .....	356
22.2.3	SPI_Slave_Init .....	357
22.2.4	SPIx_SPICR2_SPIEN_Setable .....	358
22.2.5	SPIx_SPICR2_SPIEN_Getable .....	358
22.2.6	SPIx_SPICR3_TXBFC_Clr .....	358
22.2.7	SPIx_SPICR3_RXBFC_Clr .....	358
22.2.8	SPIx_SPICR3_MERRC_Clr .....	359
22.2.9	SPIx_SPICR3_SERRC_Clr .....	359
22.2.10	SPIx_SPIIE_ERRIE_Setable .....	359



22.2.11	<i>SPIx_SPIIE_ERRIE_Getable</i> .....	359
22.2.12	<i>SPIx_SPIIE_TXIE_Setable</i> .....	360
22.2.13	<i>SPIx_SPIIE_TXIE_Getable</i> .....	360
22.2.14	<i>SPIx_SPIIE_RXIE_Setable</i> .....	360
22.2.15	<i>SPIx_SPIIE_RXIE_Getable</i> .....	361
22.2.16	<i>SPI_SSN_Set_Low</i> .....	361
22.2.17	<i>SPI_SSN_Set_High</i> .....	361
22.2.18	<i>SPI_Send_Byte</i> .....	361
22.2.19	<i>SPI_Recv_Byte</i> .....	362
22.2.20	<i>SPI_RW_Byte</i> .....	362
22.2.21	<i>SPIx_SPITXBUF_Write</i> .....	362
22.2.22	<i>SPIx_SPITXBUF_Read</i> .....	362
22.2.23	<i>SPIx_SPIRXBUF_Write</i> .....	363
22.2.24	<i>SPIx_SPIRXBUF_Read</i> .....	363
22.2.25	<i>SPIx_SPICR1_CPHOL_Set</i> .....	363
22.2.26	<i>SPIx_SPICR1_CPHOL_Get</i> .....	363
22.2.27	<i>SPIx_SPICR1_CPHA_Set</i> .....	364
22.2.28	<i>SPIx_SPICR1_CPHA_Get</i> .....	364
22.2.29	<i>SPIx_SPICR1_LSBF_Set</i> .....	364
22.2.30	<i>SPIx_SPICR1_LSBF_Get</i> .....	365
22.2.31	<i>SPIx_SPICR1_MM_Set</i> .....	365
22.2.32	<i>SPIx_SPICR1_MM_Get</i> .....	365
22.2.33	<i>SPIx_SPICR1_WAIT_Set</i> .....	366
22.2.34	<i>SPIx_SPICR1_WAIT_Get</i> .....	366
22.2.35	<i>SPIx_SPICR1_BAUD_Set</i> .....	366
22.2.36	<i>SPIx_SPICR1_BAUD_Get</i> .....	367
22.2.37	<i>SPIx_SPICR2_SSNM_Set</i> .....	367
22.2.38	<i>SPIx_SPICR2_SSNM_Get</i> .....	368
22.2.39	<i>SPIx_SPICR2_SSN_Set</i> .....	368
22.2.40	<i>SPIx_SPICR2_SSN_Get</i> .....	368
22.2.41	<i>SPIx_SPICR2_SSENSEN_Set</i> .....	369
22.2.42	<i>SPIx_SPICR2_SSENSEN_Get</i> .....	369
22.2.43	<i>SPIx_SPICR1_MSPA_Set</i> .....	369
22.2.44	<i>SPIx_SPICR1_MSPA_Get</i> .....	370
22.2.45	<i>SPIx_SPICR1_SSPA_Set</i> .....	370
22.2.46	<i>SPIx_SPICR1_SSPA_Get</i> .....	370
22.2.47	<i>SPIx_SPICR2_FLTEN_Setable</i> .....	370
22.2.48	<i>SPIx_SPICR2_FLTEN_Getable</i> .....	371
22.2.49	<i>SPIx_SPIIF_TXBE_Chk</i> .....	371
22.2.50	<i>SPIx_SPIIF_RXBF_Chk</i> .....	371



22.2.51	<i>SPIx_SPIIF_BUSY_Chr</i> .....	372
22.2.52	<i>SPIx_SPIIF_TXCOL_Clr</i> .....	372
22.2.53	<i>SPIx_SPIIF_TXCOL_Chr</i> .....	372
22.2.54	<i>SPIx_SPIIF_RXCOL_Clr</i> .....	372
22.2.55	<i>SPIx_SPIIF_RXCOL_Chr</i> .....	373
22.2.56	<i>SPIx_SPIIF_MERR_Chr</i> .....	373
22.2.57	<i>SPIx_SPIIF_SERR_Chr</i> .....	373
22.2.58	<i>SPIx_SPICR2_TXO_AC_Set</i> .....	374
22.2.59	<i>SPIx_SPICR2_TXO_AC_Get</i> .....	374
22.2.60	<i>SPIx_SPICR2_TXO_Setable</i> .....	374
22.2.61	<i>SPIx_SPICR2_TXO_Getable</i> .....	374
<b>23</b>	<b>HSPI</b> .....	<b>375</b>
23.1	HSPI 寄存器结构.....	375
23.2	HSPI 库函数.....	375
23.2.1	<i>HSPI_Deinit</i> .....	377
23.2.2	<i>HSPI_Master_Init</i> .....	377
23.2.3	<i>HSPI_Slave_Init</i> .....	378
23.2.4	<i>HSPI_SPICR2_HSPIEN_Setable</i> .....	378
23.2.5	<i>HSPI_SPICR2_HSPIEN_Getable</i> .....	379
23.2.6	<i>HSPI_SPICR3_TXBFC_Clr</i> .....	379
23.2.7	<i>HSPI_SPICR3_RXBFC_Clr</i> .....	379
23.2.8	<i>HSPI_SPICR3_MERRC_Clr</i> .....	379
23.2.9	<i>HSPI_SPICR3_SERRC_Clr</i> .....	380
23.2.10	<i>HSPI_SPIIE_ERRIE_Setable</i> .....	380
23.2.11	<i>HSPI_SPIIE_ERRIE_Getable</i> .....	380
23.2.12	<i>HSPI_SPIIE_TXIE_Setable</i> .....	381
23.2.13	<i>HSPI_SPIIE_TXIE_Getable</i> .....	381
23.2.14	<i>HSPI_SPIIE_RXIE_Setable</i> .....	381
23.2.15	<i>HSPI_SPIIE_RXIE_Getable</i> .....	381
23.2.16	<i>HSPI_SSN_Set_Low</i> .....	382
23.2.17	<i>HSPI_SSN_Set_High</i> .....	382
23.2.18	<i>HSPI_Send_Byte</i> .....	382
23.2.19	<i>HSPI_Recv_Byte</i> .....	382
23.2.20	<i>HSPI_RW_Byte</i> .....	383
23.2.21	<i>HSPI_SPITXBUF_Write</i> .....	383
23.2.22	<i>HSPI_SPITXBUF_Read</i> .....	383
23.2.23	<i>HSPI_SPIRXBUF_Write</i> .....	383
23.2.24	<i>HSPI_SPIRXBUF_Read</i> .....	384
23.2.25	<i>HSPI_SPICR1_CPHOL_Set</i> .....	384



23.2.26	<i>HSPI_SPICR1_CPHOL_Get</i> .....	384
23.2.27	<i>HSPI_SPICR1_CPHA_Set</i> .....	384
23.2.28	<i>HSPI_SPICR1_CPHA_Get</i> .....	385
23.2.29	<i>HSPI_SPICR1_LSBF_Set</i> .....	385
23.2.30	<i>HSPI_SPICR1_LSBF_Get</i> .....	385
23.2.31	<i>HSPI_SPICR1_MM_Set</i> .....	386
23.2.32	<i>HSPI_SPICR1_MM_Get</i> .....	386
23.2.33	<i>HSPI_SPICR1_WAIT_Set</i> .....	386
23.2.34	<i>HSPI_SPICR1_WAIT_Get</i> .....	387
23.2.35	<i>HSPI_SPICR1_BAUD_Set</i> .....	387
23.2.36	<i>HSPI_SPICR1_BAUD_Get</i> .....	388
23.2.37	<i>HSPI_SPICR2_SSNM_Set</i> .....	388
23.2.38	<i>HSPI_SPICR2_SSNM_Get</i> .....	388
23.2.39	<i>HSPI_SPICR2_SSN_Set</i> .....	389
23.2.40	<i>HSPI_SPICR2_SSN_Get</i> .....	389
23.2.41	<i>HSPI_SPICR2_SSNSSEN_Set</i> .....	389
23.2.42	<i>HSPI_SPICR2_SSNSSEN_Get</i> .....	389
23.2.43	<i>HSPI_SPICR1_DELAY_CFG_Set</i> .....	390
23.2.44	<i>HSPI_SPICR1_DELAY_CFG_Get</i> .....	390
23.2.45	<i>HSPI_SPICR2_FLTEN_Setable</i> .....	391
23.2.46	<i>HSPI_SPICR2_FLTEN_Getable</i> .....	391
23.2.47	<i>HSPI_SPIIF_TXBE_Chk</i> .....	391
23.2.48	<i>HSPI_SPIIF_RXBF_Chk</i> .....	392
23.2.49	<i>HSPI_SPIIF_BUSY_Chk</i> .....	392
23.2.50	<i>HSPI_SPIIF_TXCOL_Clr</i> .....	392
23.2.51	<i>HSPI_SPIIF_TXCOL_Chk</i> .....	392
23.2.52	<i>HSPI_SPIIF_RXCOL_Clr</i> .....	393
23.2.53	<i>HSPI_SPIIF_RXCOL_Chk</i> .....	393
23.2.54	<i>HSPI_SPIIF_MERR_Chk</i> .....	393
23.2.55	<i>HSPI_SPIIF_SERR_Chk</i> .....	394
23.2.56	<i>HSPI_SPICR2_TXO_AC_Set</i> .....	394
23.2.57	<i>HSPI_SPICR2_TXO_AC_Get</i> .....	394
23.2.58	<i>HSPI_SPICR2_TXO_Setable</i> .....	394
23.2.59	<i>HSPI_SPICR2_TXO_Getable</i> .....	395
<b>24</b>	<b>版本说明</b> .....	<b>396</b>

# 1 工程使用说明

所有程序的工程目录结构如下图：

名称	修改日期	类型
Device	2018/7/5 11:12	文件夹
Drivers	2018/7/5 11:12	文件夹
FM33A0XX_TesterV1.0	2018/7/5 11:12	文件夹
FM33A0XX_TesterV1.0_app	2018/7/5 11:12	文件夹
FM33A0XX_TesterV1.0_IAR	2018/7/5 11:12	文件夹
FM33A0XX_TesterV1.0_IAR_app	2018/7/5 11:12	文件夹
UserCode	2018/7/5 11:12	文件夹

- Device 文件夹下存放了 MCU 相关的文件，包含.C 文件和.H 头文件；
- Drives 文件夹下存放了工程使用到的 MCU 相关的底层模块驱动程序，包含程序.C 文件和.H 头文件；
- FM33A0XX\_TesterV1.0 文件夹下存放了 Keil 的工程设置文件，工程输出文件等；
- FM33A0XX\_TesterV1.0\_app 文件夹下存放了 Keil 的工程设置文件，工程输出文件等(bootload 用户程序)；
- FM33A0XX\_TesterV1.0\_IAR 文件夹下存放了 IAR 的工程设置文件，工程输出文件等；
- FM33A0XX\_TesterV1.0\_IAR\_app 文件夹下存放了 IAR 的工程设置文件，工程输出文件等(bootload 用户程序)；
- UserCode 文件夹下存放了主程序及用户项目程序，包含程序.C 文件和.H 头文件，用户可以在这里添加自己的程序文件；

## 2 GPIO

### 2.1 GPIO 寄存器结构:

寄存器	描述
PAINEN	GPIOx 输入使能寄存器
PBINEN	
PCINEN	
PDINEN	
PEINEN	
PFINEN	
PGINEN	
PAPUEN	GPIOx 上拉使能寄存器
PBPUEN	
PCPUEN	
PDPUEN	
PEPUEN	
PFPUEN	
PGPUEN	
PAODEN	GPIOx 开漏使能寄存器
PBODEN	
PCODEN	
PDODEN	
PEODEN	
PFODEN	
PGODEN	
PAFCR	GPIOx 功能控制寄存器
PBFCR	
PCFCR	
PDFCR	
PEFCR	
PFFCR	
PGFCR	
PADO	GPIOx 输出数据寄存器
PBDO	
PCDO	



PDDO	
PEDO	
PFDO	
PGDO	
PADSET	GPIOx 输出数据置位寄存器
PBDSET	
PCDSET	
PDDSET	
PEDSET	
PFSET	
PGDSET	
PADRESET	PIOx 输出数据复位寄存器
PBDRESET	
PCDRESET	
PDDRESET	
PEDRESET	
PFRESET	
PGDRESET	
PADIN	GPIOx 输入数据寄存器
PBDIN	
PCDIN	
PDDIN	
PEDIN	
PFIN	
PGDIN	
EXTI_SELO	外部引脚中断选择寄存器
EXTI_SEL1	
EXTI_SEL2	
EXTIOIF	外部引脚中断标志寄存器
EXTI1IF	
EXTI2IF	
FOUT_SEL	FOUT 设置寄存器
HDSEL	强驱动引脚设置寄存器
ANASEL	模拟功能选择寄存器
IODF	GPIO 输入数字滤波寄存器
PINWKEN	NWKUP 使能寄存器



## 2.2 GPIO 函数库:

序号	函数名	描述
1	GPIO_Init	GPIO 初始化
2	GPIO_ALL_Deinit	GPIO 所有寄存器初始化为缺省值
3	GPIO_Deinit	GPIO 公共寄存器初始化为缺省值
4	GPIOx_Deinit	GPIOx 非公共寄存器初始化为缺省值
5	GPIO_ReadInputDataBit	GPIOx 读取输入 pin 信号
6	GPIO_ReadOutputDataBit	GPIOx 读取输出 pin 信号
7	GPIO_ReadInputData	GPIOx 读取输入 port 信号 (32 位)
8	GPIOx_DIN_Read	GPIOx 读取输入 port 信号(低 16 位)
9	GPIO_ReadOutputData	GPIOx 读取输出 port 信号 (32 位)
10	GPIOx_DO_Read	GPIOx 读取输出 port 信号(低 16 位)
11	GPIOx_DO_Write	GPIOx 写输出 port 信号 (低 16 位)
12	GPIO_SetBits	GPIOx 输出数据置位 (32 位)
13	GPIOx_DSET_Write	GPIOx 输出 port 数据置位(低 16 位)
14	GPIO_ResetBits	GPIOx 输出数据复位 (32 位)
15	GPIOx_DRESET_Write	GPIOx 输出 port 数据复位(低 16 位)
16	GPIO_ToggleBits	GPIOx 反转输出 pin 信号
17	GPIO_Get_InitPara	获取一个 IO 口的结构体设置参数
18	GPIO_FOUTSEL_FOUTSEL_Set	FOUT 设置输出频率
19	GPIO_FOUTSEL_FOUTSEL_Get	FOUT 读取输出频率
20	GPIO_HDSEL_PG6HDEN_Setable	PG6 设置强驱动使能
21	GPIO_HDSEL_PG6HDEN_Getable	PG6 获取强驱动使能状态
22	GPIO_HDSEL_PE2HDEN_Setable	PE2 设置强驱动使能
23	GPIO_HDSEL_PE2HDEN_Getable	PE2 获取强驱动使能状态
24	GPIO_ANASEL_PE4ANS_Set	PE4 设置模拟功能
25	GPIO_ANASEL_PE4ANS_Get	PE4 获取模拟功能状态
26	GPIO_ANASEL_PE3ANS_Set	PE3 设置模拟功能
27	GPIO_ANASEL_PE3ANS_Get	PE3 获取模拟功能状态
28	GPIO_ANASEL_PC15ANS_Set	PC15 设置模拟功能
29	GPIO_ANASEL_PC15ANS_Get	PC15 获取模拟功能状态
30	GPIO_ANASEL_PC14ANS_Set	PC14 设置模拟功能
31	GPIO_ANASEL_PC14ANS_Get	PC14 获取模拟功能状态
32	GPIO_ANASEL_PC13ANS_Set	PC13 设置模拟功能
33	GPIO_ANASEL_PC13ANS_Get	PC13 获取模拟功能状态
34	GPIO_ANASEL_PC12ANS_Set	PC12 设置模拟功能

35	GPIO_ANASEL_PC12ANS_Get	PC12 获取模拟功能状态
36	GPIO_EXTI_Init	外部中断初始化
37	GPIO_EXTI_Close	外部中断禁止
38	GPIO_EXTI_Select_Pin	外部引脚选择
39	GPIO_EXTI_Select_Edge	外部中断边沿选择
40	GPIO_EXTI_EXTIxIF_ClrEx	清除外部中断标志
41	GPIO_EXTI_EXTIxIF_ChkEx	获取外部中断标志
42	GPIO_IODF_SetableEx	GPIO 输入数字滤波设置
43	GPIO_PINWKEN_SetableEx	WKUP 引脚使能

### 2.2.1 GPIO\_Init

函数名	GPIO_Init
函数原型	void GPIO_Init(GPIOx_Type* GPIOx, GPIO_InitTypeDef* para)
功能描述	GPIO 设置 pin 引脚功能
输入参数 1	GPIO 端口：PORTA~PORTG
输入参数 2	<div>GPIO 参数：</div> <div><div>pin 引脚：PIN0~PIN15</div><div>GPIO 功能： 0X00：GPIO 输入 0X01：GPIO 输出 0X02：GPIO 数字特殊功能 0X03：GPIO 模拟功能</div><div>开漏输出使能控制： 0X00：开漏关闭 0X01：开漏使能</div><div>上拉使能控制： 0X00：上拉关闭 0X01：上拉使能</div><div>输入使能控制： 0X00：输入关闭 0X01：输入使能</div></div>
输出参数	无
返回值	无

### 2.2.2 GPIO\_ALL\_Deinit

函数名	GPIO_ALL_Deinit
函数原型	GPIO 所有寄存器初始化为缺省值

功能描述	GPIO 初始化为缺省值
输入参数	无
输出参数	无
返回值	无

### 2.2.3 GPIO\_Deinit

函数名	GPIO_Deinit
函数原型	void GPIO_Deinit(void)
功能描述	GPIO 公共寄存器初始化为缺省值
输入参数	无
输出参数	无
返回值	无

### 2.2.4 GPIOx\_Deinit

函数名	GPIOx_Deinit
函数原型	void GPIOx_Deinit(GPIOx_Type* GPIOx)
功能描述	GPIOx 非公共寄存器初始化为缺省值
输入参数 1	port 端口: PORTA~PORTG
输出参数	无
返回值	无

### 2.2.5 GPIO\_ReadInputDataBit

函数名	GPIO_ReadInputDataBit
函数原型	uint8_t GPIO_ReadInputDataBit(GPIOx_Type* GPIOx, uint32_t GPIO_Pin)
功能描述	GPIOx 读取输入 pin 信号
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输出参数	无
返回值	引脚电平

### 2.2.6 GPIO\_ReadOutputDataBit

函数名	GPIO_ReadOutputDataBit
函数原型	uint8_t GPIO_ReadOutputDataBit(GPIOx_Type* GPIOx, uint32_t GPIO_Pin)
功能描述	GPIOx 读取输出 pin 信号

输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输出参数	无
返回值	引脚电平

### 2.2.7 GPIO\_ReadInputData

函数名	GPIO_ReadInputData
函数原型	uint32_t GPIO_ReadInputData(GPIOx_Type* GPIOx)
功能描述	GPIOx 读取输入 port 信号 (32 位)
输入参数 1	port 端口: PORTA~PORTG
输出参数	无
返回值	端口电平

### 2.2.8 GPIOx\_DIN\_Read

函数名	GPIOx_DIN_Read
函数原型	uint32_t GPIOx_DIN_Read(GPIOx_Type* GPIOx)
功能描述	GPIOx 读取输入 port 信号 (低 16 位)
输入参数 1	port 端口: PORTA~PORTG
输出参数	无
返回值	端口电平

### 2.2.9 GPIO\_ReadOutputData

函数名	GPIO_ReadOutputData
函数原型	uint32_t GPIO_ReadOutputData(GPIOx_Type* GPIOx)
功能描述	GPIOx 读取输出 port 信号 (32 位)
输入参数 1	port 端口: PORTA~PORTG
输出参数	无
返回值	端口电平

### 2.2.10 GPIOx\_DO\_Read

函数名	GPIOx_DO_Read
函数原型	uint32_t GPIOx_DO_Read(GPIOx_Type* GPIOx)
功能描述	GPIOx 读取输出 port 信号 (低 16 位)
输入参数 1	port 端口: PORTA~PORTG

输出参数	无
返回值	端口电平

### 2.2.11 GPIOx\_DO\_Write

函数名	GPIOx_DO_Write
函数原型	void GPIOx_DO_Write(GPIOx_Type* GPIOx, uint32_t SetValue)
功能描述	GPIOx 写输出 port 信号
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	Port 端口电平
输出参数	无
返回值	无

### 2.2.12 GPIO\_SetBits

函数名	GPIO_SetBits
函数原型	void GPIO_SetBits(GPIOx_Type* GPIOx, uint32_t GPIO_Pin)
功能描述	GPIOx 输出数据置位 (32 位)
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输出参数	无
返回值	无

### 2.2.13 GPIOx\_DSET\_Write

函数名	GPIOx_DSET_Write
函数原型	void GPIOx_DSET_Write(GPIOx_Type* GPIOx, uint32_t SetValue)
功能描述	GPIOx 输出数据置位 (低 16 位)
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	置位引脚
输出参数	无
返回值	无

### 2.2.14 GPIO\_ResetBits

函数名	GPIO_ResetBits
函数原型	void GPIO_ResetBits(GPIOx_Type* GPIOx, uint32_t GPIO_Pin)
功能描述	GPIOx 输出数据复位 (32 位)

输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输出参数	无
返回值	无

### 2.2.15 GPIOx\_DRESET\_Write

函数名	GPIOx_DRESET_Write
函数原型	void GPIOx_DRESET_Write(GPIOx_Type* GPIOx, uint32_t SetValue)
功能描述	GPIOx 输出 port 数据复位 (低 16 位)
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	复位引脚
输出参数	无
返回值	无

### 2.2.16 GPIO\_ToggleBits

函数名	GPIO_ToggleBits
函数原型	void GPIO_ToggleBits(GPIOx_Type* GPIOx, uint32_t GPIO_Pin)
功能描述	GPIOx 反转输出 pin 信号
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输出参数	无
返回值	无

### 2.2.17 GPIO\_Get\_InitPara

函数名	GPIO_Get_InitPara
函数原型	void GPIO_Get_InitPara(GPIOx_Type* GPIOx, uint32_t GPIO_Pin, GPIO_InitTypeDef* para)
功能描述	获取一个 IO 配置参数结构体
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输出参数	引脚类型: <div>             pin 引脚: PIN0~PIN15              GPIO 功能:              0X00: GPIO 输入              0X01: GPIO 输出           </div>

	0X02: GPIO 数字特殊功能 0X03: GPIO 模拟功能
	开漏输出使能控制: 0X00: 开漏关闭 0X01: 开漏使能
	上拉使能控制: 0X00: 上拉关闭 0X01: 上拉使能
	输入使能控制: 0X00: 输入关闭 0X01: 输入使能
返回值	无

### 2.2.18 GPIO\_FOUTSEL\_FOUTSEL\_Set

函数名	GPIO_FOUTSEL_FOUTSEL_Set
函数原型	void GPIO_FOUTSEL_FOUTSEL_Set(uint32_t SetValue)
功能描述	FOUT 设置输出频率
输入参数 1	FOUT 输出频率选择值: 0000: XTLF 0001: RCLP 0010: RCHF/64 0011: LSCLK 0100: AHBCLK/64 0101: RTCTM 0110: PLLO/64 0111: RTCCLK64Hz 1000: APBCLK/64 1001: LCD Booster Clock 1010: LVMOS-Monitor-RCOSC/64 1011~1111: RFU
输出参数	无
返回值	无

### 2.2.19 GPIO\_FOUTSEL\_FOUTSEL\_Get

函数名	GPIO_FOUTSEL_FOUTSEL_Get
函数原型	uint32_t GPIO_FOUTSEL_FOUTSEL_Get(void)
功能描述	FOUT 读取输出频率

输入参数 1	无
输出参数	无
返回值	FOUT 输出频率选择值: 0000: XTLF 0001: RCLP 0010: RCHF/64 0011: LSCLK 0100: AHBCLK/64 0101: RTCTM 0110: PLLO/64 0111: RTCCLK64Hz 1000: APBCLK/64 1001: LCD Booster Clock 1010: LVMOS-Monitor-RCOSC/64 1011~1111: RFU

### 2.2.20 GPIO\_HDSEL\_PG6HDEN\_Setable

函数名	GPIO_HDSEL_PG6HDEN_Setable
函数原型	void GPIO_HDSEL_PG6HDEN_Setable(FunState NewState)
功能描述	PG6 设置强驱动使能
输入参数 1	强驱动模式使能信号: ENABLE: 使能 DISABLE: 禁止
输出参数	无
返回值	无

### 2.2.21 GPIO\_HDSEL\_PE2HDEN\_Setable

函数名	GPIO_HDSEL_PE2HDEN_Setable
函数原型	void GPIO_HDSEL_PE2HDEN_Setable(FunState NewState)
功能描述	PE2 设置强驱动使能
输入参数 1	强驱动模式使能信号: ENABLE: 使能 DISABLE: 禁止
输出参数	无
返回值	无





### 2.2.22 GPIO\_HDSEL\_PG6HDEN\_Getable

函数名	GPIO_HDSEL_PG6HDEN_Getable
函数原型	FunState GPIO_HDSEL_PG6HDEN_Getable(void)
功能描述	PG6 获取强驱动使能状态
输入参数 1	无
输出参数	无
返回值	强驱动模式使能状态: ENABLE: 使能 DISABLE: 禁止

### 2.2.23 GPIO\_HDSEL\_PE2HDEN\_Getable

函数名	GPIO_HDSEL_PE2HDEN_Getable
函数原型	FunState GPIO_HDSEL_PE2HDEN_Getable(void)
功能描述	PE2 获取强驱动使能状态
输入参数 1	无
输出参数	无
返回值	强驱动模式使能状态: ENABLE: 使能 DISABLE: 禁止

### 2.2.24 GPIO\_ANASEL\_PE4ANS\_Set

函数名	GPIO_ANASEL_PE4ANS_Set
函数原型	void GPIO_ANASEL_PE4ANS_Set(uint32_t SetValue)
功能描述	PE4 设置模拟功能
输入参数 1	PE4 模拟功能选择: 0: 选择作为 ACMP1_INP1 1: 选择作为 SEG19
输出参数	无
返回值	无

### 2.2.25 GPIO\_ANASEL\_PE4ANS\_Get

函数名	GPIO_ANASEL_PE4ANS_Get
函数原型	uint32_t GPIO_ANASEL_PE4ANS_Get(void)
功能描述	PE4 设置模拟功能

输入参数 1	无
输出参数	无
返回值	PE4 模拟功能选择： 0: 选择作为 ACMP1_INP1 1: 选择作为 SEG19

### 2.2.26 GPIO\_ANASEL\_PE3ANS\_Set

函数名	GPIO_ANASEL_PE3ANS_Set
函数原型	void GPIO_ANASEL_PE3ANS_Set(uint32_t SetValue)
功能描述	PE3 设置模拟功能
输入参数 1	PE3 模拟功能选择： 0: 选择作为 ACMP1_INN1 1: 选择作为 SEG18
输出参数	无
返回值	无

### 2.2.27 GPIO\_ANASEL\_PE3ANS\_Get

函数名	GPIO_ANASEL_PE3ANS_Get
函数原型	uint32_t GPIO_ANASEL_PE3ANS_Get(void)
功能描述	PE3 设置模拟功能
输入参数 1	无
输出参数	无
返回值	PE3 模拟功能选择： 0: 选择作为 ACMP1_INN1 1: 选择作为 SEG18

### 2.2.28 GPIO\_ANASEL\_PC15ANS\_Set

函数名	GPIO_ANASEL_PC15ANS_Set
函数原型	void GPIO_ANASEL_PC15ANS_Set(uint32_t SetValue)
功能描述	PC15 设置模拟功能
输入参数 1	PC15 模拟功能选择： 0: 选择作为 ACMP1_INP0 1: 选择作为 SEG39
输出参数	无
返回值	无

**2.2.29 GPIO\_ANASEL\_PC15ANS\_Get**

函数名	GPIO_ANASEL_PC15ANS_Get
函数原型	uint32_t GPIO_ANASEL_PC15ANS_Get(void)
功能描述	PC15 设置模拟功能
输入参数 1	无
输出参数	无
返回值	PC15 模拟功能选择: 0: 选择作为 ACMP1_INP0 1: 选择作为 SEG39

**2.2.30 GPIO\_ANASEL\_PC14ANS\_Set**

函数名	GPIO_ANASEL_PC14ANS_Set
函数原型	void GPIO_ANASEL_PC14ANS_Set(uint32_t SetValue)
功能描述	PC14 设置模拟功能
输入参数 1	PC14 模拟功能选择: 0: 选择作为 ACMP1_INN0 1: 选择作为 SEG38
输出参数	无
返回值	无

**2.2.31 GPIO\_ANASEL\_PC14ANS\_Get**

函数名	GPIO_ANASEL_PC14ANS_Get
函数原型	uint32_t GPIO_ANASEL_PC14ANS_Get(void)
功能描述	PC14 设置模拟功能
输入参数 1	无
输出参数	无
返回值	PC14 模拟功能选择: 0: 选择作为 ACMP1_INN0 1: 选择作为 SEG38

**2.2.32 GPIO\_ANASEL\_PC13ANS\_Set**

函数名	GPIO_ANASEL_PC13ANS_Set
函数原型	void GPIO_ANASEL_PC13ANS_Set(uint32_t SetValue)

功能描述	PC13 设置模拟功能
输入参数 1	PC13 模拟功能选择: 0: 选择作为 ADC_IN2 1: 选择作为 SEG37
输出参数	无
返回值	无

### 2.2.33 GPIO\_ANASEL\_PC13ANS\_Get

函数名	GPIO_ANASEL_PC13ANS_Get
函数原型	uint32_t GPIO_ANASEL_PC13ANS_Get(void)
功能描述	PC13 设置模拟功能
输入参数 1	无
输出参数	无
返回值	PC13 模拟功能选择: 0: 选择作为 ADC_IN2 1: 选择作为 SEG37

### 2.2.34 GPIO\_ANASEL\_PC12ANS\_Set

函数名	GPIO_ANASEL_PC12ANS_Set
函数原型	void GPIO_ANASEL_PC12ANS_Set(uint32_t SetValue)
功能描述	PC12 设置模拟功能
输入参数 1	PC12 模拟功能选择: 0: 选择作为 ADC_IN1 1: 选择作为 SEG36
输出参数	无
返回值	无

### 2.2.35 GPIO\_ANASEL\_PC12ANS\_Get

函数名	GPIO_ANASEL_PC12ANS_Get
函数原型	uint32_t GPIO_ANASEL_PC12ANS_Get(void)
功能描述	PC12 设置模拟功能
输入参数 1	无
输出参数	无
返回值	PC12 模拟功能选择: 0: 选择作为 ADC_IN1

1: 选择作为 SEG36

### 2.2.36 GPIO\_EXTI\_Init

函数名	GPIO_EXTI_Init
函数原型	void GPIO_EXTI_Init(GPIOx_Type* GPIOx,uint32_t GPIO_Pin,GPIOExtiEdge edge_select)
功能描述	外部中断初始化
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输入参数 3	触发沿选择: 00: 上升沿 01: 下降沿 10: 两者 11: 禁止
输出参数	无
返回值	无

### 2.2.37 GPIO\_EXTI\_Close

函数名	GPIO_EXTI_Close
函数原型	void GPIO_EXTI_Close(GPIOx_Type* GPIOx,uint32_t GPIO_Pin)
功能描述	禁止外部中断
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输出参数	无
返回值	无

### 2.2.38 GPIO\_EXTI\_Select\_Pin

函数名	GPIO_EXTI_Select_Pin
函数原型	void GPIO_EXTI_Select_Pin(GPIOx_Type* GPIOx,uint32_t GPIO_Pin)
功能描述	外部引脚选择
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输出参数	无
返回值	无

### 2.2.39 GPIO\_EXTI\_Select\_Edge

函数名	GPIO_EXTI_Select_Edge
函数原型	void GPIO_EXTI_Select_Edge(GPIOx_Type* GPIOx, uint32_t GPIO_Pin, GPIOExtiEdge edge_select)
功能描述	外部中断边沿选择
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输入参数 3	触发沿选择: 00: 上升沿 01: 下降沿 10: 两者 11: 禁止
输出参数	无
返回值	无

### 2.2.40 GPIO\_EXTI\_EXTIxIF\_ClrEx

函数名	GPIO_EXTI_EXTIxIF_ClrEx
函数原型	void GPIO_EXTI_EXTIxIF_ClrEx(GPIOx_Type* GPIOx, uint32_t GPIO_Pin)
功能描述	清除外部中断标志
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输出参数	无
返回值	无

### 2.2.41 GPIO\_EXTI\_EXTIxIF\_ChkEx

函数名	GPIO_EXTI_EXTIxIF_ChkEx
函数原型	FlagStatus GPIO_EXTI_EXTIxIF_ChkEx(GPIOx_Type* GPIOx, uint32_t GPIO_Pin)
功能描述	获取外部中断标志
输入参数 1	port 端口: PORTA~PORTG
输入参数 2	pin 端口内引脚: PIN0~PIN15
输出参数	无
返回值	外部中断标志: DISABLE: 未发生外部中断 ENABLE: 发生外部中断

### 2.2.42 GPIO\_IODF\_SetableEx

函数名	GPIO_IODF_SetableEx
函数原型	void GPIO_IODF_SetableEx(uint32_t DFPinDef, FunState NewState)
功能描述	GPIO 输入数字滤波设置
输入参数 1	数字滤波 PIN 引脚
输入参数 2	数字滤波使能状态： DISABLE：禁止数字滤波 ENABLE：使能数字滤波
输出参数	无
返回值	无

### 2.2.43 GPIO\_PINWKEN\_SetableEx

函数名	GPIO_PINWKEN_SetableEx
函数原型	void GPIO_PINWKEN_SetableEx(uint32_t NWKPinDef, FunState NewState)
功能描述	NWKUP 引脚使能
输入参数 1	NWKUP PIN 引脚
输入参数 2	NWKUP 引脚使能状态： DISABLE：禁止 NWKUP ENABLE：使能 NWKUP
输出参数	无
返回值	无



## 3 RCC

### 3.1 RCC 寄存器结构

寄存器	描述
RSTCFG	复位配置寄存器
SOFT_RST	软件复位寄存器
RSTFLAG	复位标志寄存器
SYSCLKSEL	系统时钟设置寄存器
RCHFCON	RCHF 控制寄存器
RCHFTRIM	RCHF 调校寄存器
PLLCON	PLL 控制寄存器
RCLPCON	RCLP 控制寄存器
RCLPTRIM	RCLP 调校寄存器
XTLFIW	XTLF 振荡强度设置寄存器
PERCLKCON1	外设时钟控制寄存器 1
PERCLKCON2	外设时钟控制寄存器 2
PERCLKCON3	外设时钟控制寄存器 3
PERCLKCON4	外设时钟控制寄存器 4
MPRIL	AHB Master 优先级设置寄存器

### 3.2 RCC 库函数

序号	函数名	描述
1	RCC_SysClk_Init	系统时钟初始化
2	RCC_RCHF_Init	RCHF 时钟初始化
3	RCC_PLL_Init	PLL 锁相环初始化
4	RCC_Deinit	芯片时钟初始化为缺省值
5	RCC_PERCLK_SetableEx	设置外设时钟使能
6	RCC_Init_RCHF_Trim	设置 RCHF 常温校准值载入
7	RCC_GetClocksFreq	获取芯片已设置的不同时钟频率
8	RCC_SYSCLKSEL_SYSCLKSEL_Set	系统时钟源选择
9	RCC_SYSCLKSEL_SYSCLKSEL_Get	获取系统时钟源选择的设置
10	RCC_RCHFCON_FSEL_Set	设置 RCHF 时钟频率选择
11	RCC_RCHFCON_FSEL_Get	获取 RCHF 时钟频率选择的设置



12	RCC_RCHFCON_RCHFEN_Setable	设置 RCHF 使能
13	RCC_RCHFCON_RCHFEN_Getable	获取 RCHF 使能状态
14	RCC_RCHFTRIM_Write	设置 RCHF 频率调校
15	RCC_RCHFTRIM_Read	获取 RCHF 频率调校设置值
16	RCC_SYSCLKSEL_LPM_RCLP_OFF_Setable	休眠模式下设置 RCLP
17	RCC_SYSCLKSEL_LPM_RCLP_OFF_Getable	获取休眠模式下对 RCLP 的设置
18	RCC_RCLPTRIM_Write	设置 RCLP 频率调校
19	RCC_RCLPTRIM_Read	获取 RCLP 频率调校设置值
20	RCC_RCLPCON_RCLP_EN_B_Chk	设置 RCLP 使能选择
21	RCC_XTLFIPW_XTLFIPW_Set	设置 XTLF 工作电流选择
22	RCC_XTLFIPW_XTLFIPW_Get	获取 XTLF 工作电流选择设定
23	RCC_PLLCON_PLLDB_Set	设置 PLL 倍频数
24	RCC_PLLCON_PLLDB_Get	获取 PLL 倍频值的设定
25	RCC_PLLCON_PLLOSEL_Set	设置 PLL 输出选择
26	RCC_PLLCON_PLLOSEL_Get	获取 PLL 输出选择设定
27	RCC_PLLCON_PLLINSEL_Set	设置 PLL 输入选择
28	RCC_PLLCON_PLLINSEL_Get	获取 PLL 输入选择设定
29	RCC_PLLCON_PLEN_Setable	设置 PLL 使能设定
30	RCC_PLLCON_PLEN_Getable	获取 PLL 使能状态
31	RCC_SYSCLKSEL_AHBPRES_Set	设置 AHB 时钟分频选择
32	RCC_SYSCLKSEL_AHBPRES_Get	获取 AHB 时钟分频设定
33	RCC_MPRIL_MPRIL_Set	设置 AHB Master 优先级设置
34	RCC_MPRIL_MPRIL_Get	获取 AHB Master 优先级设置值
35	RCC_SYSCLKSEL_APBPRES_Set	设置 APB 时钟分频选择
36	RCC_SYSCLKSEL_APBPRES_Get	获取 APB 时钟分频设定
37	RCC_SYSCLKSEL_EXTICKSEL_Set	设置 EXTI 中断与数字滤波采样时钟选择
38	RCC_SYSCLKSEL_EXTICKSEL_Get	获取 EXTI 中断与数字滤波采样时钟选择设定
39	RCC_SYSCLKSEL_SLP_ENEXTI_Setable	设置休眠模式下 EXTI 采样
40	RCC_SYSCLKSEL_SLP_ENEXTI_Getable	获取休眠模式下 EXTI 采样设置值
41	RCC_PERCLKCON2_ADCCCKSEL_Set	设置 ADC 工作时钟选择
42	RCC_PERCLKCON2_ADCCCKSEL_Get	获取 ADC 工作时钟选择值

### 3.2.1 RCC\_SysClk\_Init

函数名	RCC_SysClk_Init
函数原型	void RCC_SysClk_Init(RCC_SYSCLK_InitTypeDef* para)
功能描述	系统时钟初始化

输入参数 1	<p>系统时钟设置的相关参数：</p> <p>系统时钟源选择：</p> <p>00: RCHF 01: RCLP 10: LSCLK 11: PLL</p> <p>AHB 时钟分频选择：</p> <p>0XX: 不分频 100: 2 分频 101: 4 分频 110: 8 分频 111: 16 分频</p> <p>APB 时钟分频选择：</p> <p>0XX: 不分频 100: 2 分频 101: 4 分频 110: 8 分频 111: 16 分频</p> <p>EXTI 中断采样、IO 数字滤波时钟选择：</p> <p>0: 外部引脚中断使用 AHBCLK 采样 1: 外部引脚中断使用 LSCLK 采样</p> <p>睡眠模式下 EXTI 采样设置：</p> <p>0: 休眠模式下禁止外部引脚中断采样 1: 休眠模式下使能外部引脚中断采样</p> <p>睡眠模式下 PCLP 设置：</p> <p>0: 休眠模式下保持 RCLP 开启 1: 休眠模式下禁止 RCLP</p>
输出参数	无
返回值	无

### 3.2.2 RCC\_RCHF\_Init

函数名	RCC_RCHF_Init
函数原型	void RCC_RCHF_Init(RCC_RCHF_InitTypeDef* para)
功能描述	RCHF 时钟初始化
输入参数 1	<p>RCHF 时钟设置的相关参数：</p> <p>RCHF 时钟频率选择：</p> <p>00: 8MHz 01: 16MHz 10: 24MHz 11: RFU</p> <p>RCHF 使能：</p>



	DISABLE: 禁止 RCHF ENABLE: 使能 RCHF
输出参数	无
返回值	无

### 3.2.3 RCC\_PLL\_Init

函数名	RCC_PLL_Init
函数原型	void RCC_PLL_Init(RCC_PLL_InitTypeDef* para)
功能描述	PLL 初始化
输入参数 1	PLL 设置的相关参数: PLL 倍频比: 0~1023 输出选择: 0: 选择 PLL 一倍输出作为数字电路内的 PLL 时钟 1: 选择 PLL 二倍输出作为数字电路内的 PLL 时钟 PLL 输入源选择: 00/11: XTLEF 01: RCLP 10: RCHF/256 PLL 使能控制: DISABLE: 禁止 PLL ENABLE: 使能 PLL
输出参数	无
返回值	无

### 3.2.4 RCC\_Deinit

函数名	RCC_Deinit
函数原型	void RCC_Deinit(void)
功能描述	芯片时钟初始化为缺省值
输入参数 1	无
输出参数	无
返回值	无

### 3.2.5 RCC\_PERCLK\_SetableEx

函数名	RCC_PERCLK_SetableEx
函数原型	void RCC_PERCLK_SetableEx(uint32_t periph_def, FunState NewState)
功能描述	设置外设时钟使能

输入参数 1	外设模块标号: 0x01XXXXXX~0x04XXXXXX
输入参数 2	外设时钟使能状态: DISABLE: 禁止外设时钟 ENABLE: 使能外设时钟
输出参数	无
返回值	无

### 3.2.6 RCC\_Init\_RCHF\_Trim

函数名	RCC_Init_RCHF_Trim
函数原型	void RCC_Init_RCHF_Trim( uint8_t ClkMode )
功能描述	设置 RCHF 常温校准值载入
输入参数 1	RCHF 频率设置: 0X01: 8M 0X02: 16M 0X03: 24M 0X04: 32M 其他: 8M
输出参数	无
返回值	无

### 3.2.7 RCC\_GetClocksFreq

函数名	RCC_GetClocksFreq					
函数原型	void RCC_GetClocksFreq(RCC_ClocksType* para)					
功能描述	获取芯片已设置的不同时钟频率					
输入参数 1	无					
输出参数	<div>不同时钟的频率参数：<table><tr><td>SYSCLK_Frequency: 系统时钟频率</td></tr><tr><td>AHBCLK_Frequency: AHB 时钟频率</td></tr><tr><td>APBCLK_Frequency: APB 时钟频率</td></tr><tr><td>RCHF_Frequency: RCHF 时钟频率</td></tr><tr><td>PLL_Frequency: PLL 时钟频率</td></tr></table></div>	SYSCLK_Frequency: 系统时钟频率	AHBCLK_Frequency: AHB 时钟频率	APBCLK_Frequency: APB 时钟频率	RCHF_Frequency: RCHF 时钟频率	PLL_Frequency: PLL 时钟频率
SYSCLK_Frequency: 系统时钟频率						
AHBCLK_Frequency: AHB 时钟频率						
APBCLK_Frequency: APB 时钟频率						
RCHF_Frequency: RCHF 时钟频率						
PLL_Frequency: PLL 时钟频率						
返回值	无					

### 3.2.8 RCC\_SYSCLKSEL\_SYSCLKSEL\_Set

函数名	RCC_SYSCLKSEL_SYSCLKSEL_Set
-----	-----------------------------

函数原型	void RCC_SYSCLKSEL_SYSCLKSEL_Set(uint32_t SetValue)
功能描述	设置系统时钟源选择
输入参数 1	系统时钟源选择的设置： 00: RCHF 01: RCLP 10: LSCLK 11: PLL
输出参数	无
返回值	无

### 3.2.9 RCC\_SYSCLKSEL\_SYSCLKSEL\_Get

函数名	RCC_SYSCLKSEL_SYSCLKSEL_Get
函数原型	uint32_t RCC_SYSCLKSEL_SYSCLKSEL_Get(void)
功能描述	获取系统时钟源选择的设置
输入参数 1	无
输出参数	无
返回值	系统时钟源选择的设置： 00: RCHF 01: RCLP 10: LSCLK 11: PLL

### 3.2.10 RCC\_RCHFCON\_FSEL\_Set

函数名	RCC_RCHFCON_FSEL_Set
函数原型	void RCC_RCHFCON_FSEL_Set(uint32_t SetValue)
功能描述	设置 RCHF 时钟频率选择
输入参数 1	RCHF 时钟频率选择设置： 00: 8MHz 01: 16MHz 10: 24MHz 11: RFU
输出参数	无
返回值	无

### 3.2.11 RCC\_RCHFCON\_FSEL\_Get

函数名	RCC_RCHFCON_FSEL_Get
函数原型	uint32_t RCC_RCHFCON_FSEL_Get(void)
功能描述	获取 RCHF 时钟频率选择的设置
输入参数 1	无
输出参数	无
返回值	RCHF 时钟频率选择设置： 00: 8MHz 01: 16MHz 10: 24MHz 11: RFU

### 3.2.12 RCC\_RCHFCON\_RCHFEN\_Setable

函数名	RCC_RCHFCON_RCHFEN_Setable
函数原型	void RCC_RCHFCON_RCHFEN_Setable(FunState NewState)
功能描述	设置 RCHF 使能
输入参数 1	RCHF 使能状态： DISABLE: 禁止 RCHF 时钟 ENABLE: 使能 RCHF 时钟
输出参数	无
返回值	无

### 3.2.13 RCC\_RCHFCON\_RCHFEN\_Getable

函数名	RCC_RCHFCON_RCHFEN_Getable
函数原型	FunState RCC_RCHFCON_RCHFEN_Getable(void)
功能描述	获取 RCHF 使能状态
输入参数 1	无
输出参数	无
返回值	RCHF 使能状态： DISABLE: 禁止 RCHF 时钟 ENABLE: 使能 RCHF 时钟

### 3.2.14 RCC\_RCHFTRIM\_Write

函数名	RCC_RCHFTRIM_Write
-----	--------------------



函数原型	void RCC_RCHFTRIM_Write(uint32_t SetValue)
功能描述	设置 RCHF 频率调校
输入参数 1	RCHF 频率调校的设置数值：0X00~0X3F
输出参数	无
返回值	无

### 3.2.15 RCC\_RCHFTRIM\_Read

函数名	RCC_RCHFTRIM_Read
函数原型	uint32_t RCC_RCHFTRIM_Read(void)
功能描述	获取 RCHF 频率调校设置值
输入参数 1	无
输出参数	无
返回值	RCHF 频率调校的设置数值： 0X00~0X3F

### 3.2.16 RCC\_SYSCLKSEL\_LPM\_RCLP\_OFF\_Setable

函数名	RCC_SYSCLKSEL_LPM_RCLP_OFF_Setable
函数原型	void RCC_SYSCLKSEL_LPM_RCLP_OFF_Setable(FunState NewState)
功能描述	设置休眠模式下对 RCLP 操作
输入参数 1	休眠模式下 RCLP 使能状态： DISABLE：禁止 RCLP 时钟 ENABLE：使能 RCLP 时钟
输出参数	无
返回值	无

### 3.2.17 RCC\_SYSCLKSEL\_LPM\_RCLP\_OFF\_Getable

函数名	RCC_SYSCLKSEL_LPM_RCLP_OFF_Getable
函数原型	FunState RCC_SYSCLKSEL_LPM_RCLP_OFF_Getable(void)
功能描述	获取休眠模式下对 RCLP 的设置
输入参数 1	无
输出参数	无
返回值	休眠模式下 RCLP 使能状态： DISABLE：禁止 RCLP 时钟 ENABLE：使能 RCLP 时钟

### 3.2.18 RCC\_RCLPTRIM\_Write

函数名	RCC_RCLPTRIM_Write
函数原型	void RCC_RCLPTRIM_Write(uint32_t SetValue)
功能描述	设置 RCLP 频率调校
输入参数 1	RCLP 频率调校值: 0-F
输出参数	无
返回值	无

### 3.2.19 RCC\_RCLPTRIM\_Read

函数名	RCC_RCLPTRIM_Read
函数原型	uint32_t RCC_RCLPTRIM_Read(void)
功能描述	获取 RCLP 频率调校设置值
输入参数 1	无
输出参数	无
返回值	RCLP 频率调校值:0-F

### 3.2.20 RCC\_RCLPCON\_RCLP\_EN\_B\_Chk

函数名	RCC_RCLPCON_RCLP_EN_B_Chk
函数原型	FlagStatus RCC_RCLPCON_RCLP_EN_B_Chk(void)
功能描述	获取 RCLP 使能状态
输入参数 1	无
输出参数	无
返回值	RCLP 使能状态: DISABLE:关闭 RCLP ENABLE:使能 RCLP

### 3.2.21 RCC\_XTLFIPW\_XTLFIPW\_Set

函数名	RCC_XTLFIPW_XTLFIPW_Set
函数原型	void RCC_XTLFIPW_XTLFIPW_Set(uint32_t SetValue)
功能描述	设置 XTLF 工作电流选择
输入参数 1	XTLF 工作电流: 000: 450nA 001: 400nA 010: 350nA



	011: 300nA 100: 250nA 101: 200nA 110: 150nA 111: 100nA
输出参数	无
返回值	无

### 3.2.22 RCC\_XTLFIPW\_XTLFIPW\_Get

函数名	RCC_XTLFIPW_XTLFIPW_Get
函数原型	uint32_t RCC_XTLFIPW_XTLFIPW_Get(void)
功能描述	获取 XTLF 工作电流选择设定
输入参数 1	无
输出参数	无
返回值	XTLF 工作电流: 000: 450nA 001: 400nA 010: 350nA 011: 300nA 100: 250nA 101: 200nA 110: 150nA 111: 100nA

### 3.2.23 RCC\_PLLCON\_PLLDB\_Set

函数名	RCC_PLLCON_PLLDB_Set
函数原型	void RCC_PLLCON_PLLDB_Set(uint32_t SetValue)
功能描述	设置 PLL 倍频数设置
输入参数 1	PLL 倍频数: 0-3FFFF
输出参数	无
返回值	无

### 3.2.24 RCC\_PLLCON\_PLLDB\_Get

函数名	RCC_PLLCON_PLLDB_Get
函数原型	uint32_t RCC_PLLCON_PLLDB_Get(void)



功能描述	获取 PLL 倍频值的设定
输入参数 1	无
输出参数	无
返回值	PLL 倍频数: 0-3FFFF

### 3.2.25 RCC\_PLLCON\_PLLOSEL\_Set

函数名	RCC_PLLCON_PLLOSEL_Set
函数原型	void RCC_PLLCON_PLLOSEL_Set(uint32_t SetValue)
功能描述	设置 PLL 输出选择
输入参数 1	PLL 输出选择: 0: 选择 PLL 一倍输出作为数字电路内的 PLL 时钟 1: 选择 PLL 两倍输出作为数字电路内的 PLL 时钟
输出参数	无
返回值	无

### 3.2.26 RCC\_PLLCON\_PLLOSEL\_Get

函数名	RCC_PLLCON_PLLOSEL_Get
函数原型	uint32_t RCC_PLLCON_PLLOSEL_Get(void)
功能描述	获取 PLL 输出选择设定
输入参数 1	无
输出参数	无
返回值	PLL 输出选择: 0: 选择 PLL 一倍输出作为数字电路内的 PLL 时钟 1: 选择 PLL 两倍输出作为数字电路内的 PLL 时钟

### 3.2.27 RCC\_PLLCON\_PLLINSEL\_Set

函数名	RCC_PLLCON_PLLINSEL_Set
函数原型	void RCC_PLLCON_PLLINSEL_Set(uint32_t SetValue)
功能描述	设置 PLL 输入选择
输入参数 1	PLL 输入源选择: 00/11: XTUF 01: RCLP 10: RCHF/256
输出参数	无
返回值	无

### 3.2.28 RCC\_PLLCON\_PLLINSEL\_Get

函数名	RCC_PLLCON_PLLINSEL_Get
函数原型	uint32_t RCC_PLLCON_PLLINSEL_Get(void)
功能描述	获取 PLL 输入选择设定
输入参数 1	无
输出参数	无
返回值	PLL 输入源选择: 00/11: XTLF 01: RCLP 10: RCHF/256

### 3.2.29 RCC\_PLLCON\_PPLEN\_Setable

函数名	RCC_PLLCON_PPLEN_Setable
函数原型	void RCC_PLLCON_PPLEN_Setable(FunState NewState)
功能描述	设置 PLL 使能
输入参数 1	PLL 使能状态 DISABLE: 禁止 PLL ENABLE: 使能 PLL
输出参数	无
返回值	无

### 3.2.30 RCC\_PLLCON\_PPLEN\_Getable

函数名	RCC_PLLCON_PPLEN_Getable
函数原型	FunState RCC_PLLCON_PPLEN_Getable(void)
功能描述	获取 PLL 使能状态
输入参数 1	无
输出参数	无
返回值	PLL 使能状态 DISABLE: 禁止 PLL ENABLE: 使能 PLL

### 3.2.31 RCC\_SYSCLKSEL\_AHBPRES\_Set

函数名	RCC_SYSCLKSEL_AHBPRES_Set
-----	---------------------------

函数原型	void RCC_SYSCLKSEL_AHBPRES_Set(uint32_t SetValue)
功能描述	设置 AHB 时钟分频选择
输入参数 1	AHB 时钟分频选择: 0XX: 不分频 100: 2 分频 101: 4 分频 110: 8 分频 111: 16 分频
输出参数	无
返回值	无

### 3.2.32 RCC\_SYSCLKSEL\_AHBPRES\_Get

函数名	RCC_SYSCLKSEL_AHBPRES_Get
函数原型	uint32_t RCC_SYSCLKSEL_AHBPRES_Get(void)
功能描述	获取 AHB 时钟分频设定
输入参数 1	无
输出参数	无
返回值	AHB 时钟分频选择: 0XX: 不分频 100: 2 分频 101: 4 分频 110: 8 分频 111: 16 分频

### 3.2.33 RCC\_MPRIL\_MPRIL\_Set

函数名	RCC_MPRIL_MPRIL_Set
函数原型	void RCC_MPRIL_MPRIL_Set(uint32_t SetValue)
功能描述	设置 AHB Master 优先级设置
输入参数 1	AHB Master 优先级设置选择: 0: DMA 选择 1: CPU 选择
输出参数	无
返回值	无

**3.2.34 RCC\_MPRIL\_MPRIL\_Get**

函数名	RCC_MPRIL_MPRIL_Get
函数原型	uint32_t RCC_MPRIL_MPRIL_Get(void)
功能描述	获取 AHB Master 优先级设置值
输入参数 1	无
输出参数	无
返回值	AHB Master 优先级设置选择: 0: DMA 选择 1: CPU 选择

**3.2.35 RCC\_SYCLKSEL\_APBPRES\_Set**

函数名	RCC_SYCLKSEL_APBPRES_Set
函数原型	void RCC_SYCLKSEL_APBPRES_Set(uint32_t SetValue)
功能描述	设置 APB 时钟分频选择
输入参数 1	APB 时钟分频选择: 0XX: 不分频 100: 2 分频 101: 4 分频 110: 8 分频 111: 16 分频
输出参数	无
返回值	无

**3.2.36 RCC\_SYCLKSEL\_APBPRES\_Get**

函数名	RCC_SYCLKSEL_APBPRES_Get
函数原型	uint32_t RCC_SYCLKSEL_APBPRES_Get(void)
功能描述	获取 APB 时钟分频设定
输入参数 1	无
输出参数	无
返回值	APB 时钟分频选择: 0XX: 不分频 100: 2 分频 101: 4 分频 110: 8 分频 111: 16 分频



### 3.2.37 RCC\_SYSCCLKSEL\_EXTICKSEL\_Set

函数名	RCC_SYSCCLKSEL_EXTICKSEL_Set
函数原型	void RCC_SYSCCLKSEL_EXTICKSEL_Set(uint32_t SetValue)
功能描述	设置 EXTI 中断与数字滤波采样时钟选择
输入参数 1	EXTI 中断采样、IO 数字滤波时钟选择： 0：外部引脚中断使用 AHBCLK 采样 1：外部引脚中断使用 LSCLK 采样
输出参数	无
返回值	无

### 3.2.38 RCC\_SYSCCLKSEL\_EXTICKSEL\_Get

函数名	RCC_SYSCCLKSEL_EXTICKSEL_Get
函数原型	uint32_t RCC_SYSCCLKSEL_EXTICKSEL_Get(void)
功能描述	获取 EXTI 中断与数字滤波采样时钟选择设定
输入参数 1	无
输出参数	无
返回值	EXTI 中断采样、IO 数字滤波时钟选择： 0：外部引脚中断使用 AHBCLK 采样 1：外部引脚中断使用 LSCLK 采样

### 3.2.39 RCC\_SYSCCLKSEL\_SLP\_ENEXTI\_Setable

函数名	RCC_SYSCCLKSEL_SLP_ENEXTI_Setable
函数原型	void RCC_SYSCCLKSEL_SLP_ENEXTI_Setable(FunState NewState)
功能描述	设置休眠模式下 EXTI 采样
输入参数 1	睡眠模式下 EXTI 采样设置： 0：休眠模式下禁止外部引脚中断采样 1：休眠模式下使能外部引脚中断采样
输出参数	无
返回值	无

### 3.2.40 RCC\_SYSCCLKSEL\_SLP\_ENEXTI\_Getable

函数名	RCC_SYSCCLKSEL_SLP_ENEXTI_Getable
函数原型	FunState RCC_SYSCCLKSEL_SLP_ENEXTI_Getable(void)

功能描述	获取休眠模式下 EXTI 采样设置
输入参数 1	无
输出参数	无
返回值	睡眠模式下 EXTI 采样设置： 0：休眠模式下禁止外部引脚中断采样 1：休眠模式下使能外部引脚中断采样

### 3.2.41 RCC\_PERCLKCON2\_ADCCCKSEL\_Set

函数名	RCC_PERCLKCON2_ADCCCKSEL_Set
函数原型	void RCC_PERCLKCON2_ADCCCKSEL_Set(uint32_t SetValue)
功能描述	设置 ADC 工作时钟选择
输入参数 1	ADC 工作时钟选择： 000: RCHF 001: RCHF/2 010: RCHF/4 011: RCHF/8 100: RCHF/16 101: RCHF/32 110: RCHF/64 111: RCLP
输出参数	无
返回值	无

### 3.2.42 RCC\_PERCLKCON2\_ADCCCKSEL\_Get

函数名	RCC_PERCLKCON2_ADCCCKSEL_Get
函数原型	uint32_t RCC_PERCLKCON2_ADCCCKSEL_Get(void)
功能描述	获取 ADC 工作时钟选择值
输入参数 1	无
输出参数	无
返回值	ADC 工作时钟选择： 000: RCHF 001: RCHF/2 010: RCHF/4 011: RCHF/8 100: RCHF/16 101: RCHF/32



110: RCHF/64

111: RCLP

## 4 PMU

### 4.1 PMU 寄存器结构

寄存器	描述
LPMCFG	低功耗控制寄存器
WKDLYCON	唤醒时间控制寄存器
WKPFLAG	引脚唤醒源标志寄存器
LPREIE	PMU 中断使能寄存器
LPREIF	PMU 中断标志寄存器

### 4.2 PMU 函数库

序号	函数名	描述
1	PMU_SleepCfg_Init	低功耗模式初始化
2	PMU_Deinit	PMU 初始化为缺省值
3	PMU_LPMCFG_PMOD_Set	设置低功耗模式
4	PMU_LPMCFG_PMOD_Get	获取低功耗模式设置
5	PMU_LPMCFG_CVS_Setable	设置低功耗模式内核电压开关
6	PMU_LPMCFG_CVS_Getable	获取低功耗模式内核电压开关设置
7	PMU_LPREIE_LPREIE_Setable	设置 LP Run 错误中断使能
8	PMU_LPREIE_LPREIE_Getable	获取 LP Run 错误中断使能状态
9	PMU_LPREIF_LPREIF_Clr	清除 LP Run 错误中断标志
10	PMU_LPREIF_LPREIF_Chk	获取 LP Run 错误中断标志
11	PMU_LPMCFG_SLPDP_Set	设置是否发生 DeepSleep
12	PMU_LPMCFG_SLPDP_Get	获取是否发生 DeepSleep 的设置
13	PMU_LPMCFG_DSLPRAM_EXT_Set	设置 DeepSleep 模式下 RAM 数据保持
14	PMU_LPMCFG_DSLPRAM_EXT_Get	获取 DeepSleep 模式下 RAM 数据保持设置
15	MU_LPREIE_SLPEIE_Setable	设置 Sleep 错误中断使能
16	MU_LPREIE_SLPEIE_Getable	获取 Sleep 错误中断使能状态
17	PMU_LPREIF_SLPEIF_Clr	清除 Sleep 错误中断标志



18	PMU_LPREIF_SLPEIF_Chk	获取 Sleep 错误中断标志
19	PMU_WKDLYCON_TRHR_Set	设置 BG 使能到开启 LDO 并开始读取 Flash 的时间
20	PMU_WKDLYCON_TRHR_Get	获取 BG 使能到开启 LDO 并开始读取 Flash 的时间设置
21	PMU_WKPFLAG_DBGWKF_Clr	清除 CPU Debugger 唤醒标志
22	PMU_WKPFLAG_DBGWKF_Chk	获取 CPU Debugger 唤醒标志
23	PMU_WKPFLAG_WKPx_F_ClrEx	清除 NWKUPx Pin 唤醒标志
24	PMU_WKPFLAG_WKPx_F_ChkEx	获取 NWKUPx Pin 唤醒标志
25	PMU_LPMC_CFG_XTOFF_Set	设置 XTLF 使能
26	PMU_LPMC_CFG_XTOFF_Get	获取 XTLF 使能状态
27	PMU_LPMC_CFG_LDO15EN_Chk	获取 LDO15 使能状态

#### 4.2.1 PMU\_SleepCfg\_Init

函数名	PMU_SleepCfg_Init
函数原型	void PMU_SleepCfg_Init(PMU_SleepCfg_InitTypeDef* SleepCfg_InitStruct)
功能描述	低功耗模式初始化
输入参数 1	<div>低功耗模式参数：</div> <div><div>低功耗模式设置： 00: Active mode 01: LPRUN mode 10: Sleep mode 11: RFU</div><div>DeepSleep 控制： 0: 常规 Sleep 模式 1: DeepSleep 模式</div><div>DeepSleep 模式下 RAM 数据保持设置： 0: DeepSleep 模式下保持所有 RAM 数据 1: DeepSleep 模式下保持 16KB RAM 数据</div><div>CVS (CoreVoltageScaling) 设置： 0: 低功耗模式下不能能内核电压调整 1: 低功耗模式下降低内核电压</div><div>禁止 XTLF, 仅在 Sleep/DeepSleep 下起作用： 0: 保持 XTLF 开启 1: 禁止 XTLF</div><div>M0 系统控制寄存器：一般设置为 0</div></div>
输出参数	无
返回值	无



#### 4.2.2 PMU\_Deinit

函数名	PMU_Deinit
函数原型	void PMU_Deinit(void)
功能描述	PMU 初始化为缺省值
输入参数 1	无
输出参数	无
返回值	无

#### 4.2.3 PMU\_LPMCFG\_PMOD\_Set

函数名	PMU_LPMCFG_PMOD_Set
函数原型	void PMU_LPMCFG_PMOD_Set(uint32_t SetValue)
功能描述	设置低功耗模式
输入参数 1	低功耗模式: 00: Active mode 01: LPRUN mode 10: Sleep mode 11: RFU
输出参数	无
返回值	无

#### 4.2.4 PMU\_LPMCFG\_PMOD\_Get

函数名	PMU_LPMCFG_PMOD_Get
函数原型	uint32_t PMU_LPMCFG_PMOD_Get(void)
功能描述	获取低功耗模式设置
输入参数 1	无
输出参数	无
返回值	低功耗模式: 00: Active mode 01: LPRUN mode 10: Sleep mode 11: RFU

#### 4.2.5 PMU\_LPMCFG\_CVS\_Setable

函数名	PMU_LPMCFG_CVS_Setable
-----	------------------------

函数原型	void PMU_LPMCFG_CVS_Setable(FunState NewState)
功能描述	设置低功耗模式内核电压开关
输入参数 1	CVS (CoreVoltageScaling) : 0: 低功耗模式下不能能内核电压调整 1: 低功耗模式下降低内核电压
输出参数	无
返回值	无

#### 4.2.6 PMU\_LPMCFG\_CVS\_Getable

函数名	PMU_LPMCFG_CVS_Getable
函数原型	FunState PMU_LPMCFG_CVS_Getable(void)
功能描述	获取低功耗模式内核电压开关设置
输入参数 1	无
输出参数	无
返回值	CVS (CoreVoltageScaling) 设置: 0: 低功耗模式下不能能内核电压调整 1: 低功耗模式下降低内核电压

#### 4.2.7 PMU\_LPREIE\_LPREIE\_Setable

函数名	PMU_LPREIE_LPREIE_Setable
函数原型	void PMU_LPREIE_LPREIE_Setable(FunState NewState)
功能描述	设置 LP Run 错误中断使能
输入参数 1	LP Run 错误中断使能状态: DISABLE: 禁止 LP Run 错误中断 ENABLE: 使能 LP Run 错误中断
输出参数	无
返回值	无

#### 4.2.8 PMU\_LPREIE\_LPREIE\_Getable

函数名	PMU_LPREIE_LPREIE_Getable
函数原型	FunState PMU_LPREIE_LPREIE_Getable(void)
功能描述	获取 LP Run 错误中断使能状态
输入参数 1	无
输出参数	无
返回值	LP Run 错误中断使能状态:



DISABLE: 禁止 LP Run 错误中断  
ENABLE: 使能 LP Run 错误中断

#### 4.2.9 PMU\_LPREIF\_LPREIF\_Clr

函数名	PMU_LPREIF_LPREIF_Clr
函数原型	void PMU_LPREIF_SLPEIF_Clr(void)
功能描述	清除 LP Run 错误中断标志
输入参数 1	无
输出参数	无
返回值	无

#### 4.2.10 PMU\_LPREIF\_LPREIF\_Chk

函数名	PMU_LPREIF_LPREIF_Chk
函数原型	FlagStatus PMU_LPREIF_LPREIF_Chk(void)
功能描述	获取 LP Run 错误中断标志
输入参数 1	无
输出参数	无
返回值	LP Run 错误中断使能标志: 0: 未发生 LP Run 错误中断 1: 发生 LP Run 错误中断

#### 4.2.11 PMU\_LPMCFCG\_SLPDP\_Set

函数名	PMU_LPMCFCG_SLPDP_Set
函数原型	void PMU_LPMCFCG_SLPDP_Set(uint32_t SetValue)
功能描述	设置是否发生 DeepSleep
输入参数 1	DeepSleep 控制: 0: 常规 Sleep 模式 1: DeepSleep 模式
输出参数	无
返回值	无

#### 4.2.12 PMU\_LPMCFCG\_SLPDP\_Get

函数名	PMU_LPMCFCG_SLPDP_Get
函数原型	uint32_t PMU_LPMCFCG_SLPDP_Get(void)



功能描述	获取是否发生 DeepSleep 的设置
输入参数 1	无
输出参数	无
返回值	DeepSleep 控制： 0：常规 Sleep 模式 1：DeepSleep 模式

#### 4.2.13 PMU\_LPMC\_CFG\_DSLPRAM\_EXT\_Set

函数名	PMU_LPMC_CFG_DSLPRAM_EXT_Set
函数原型	void PMU_LPMC_CFG_DSLPRAM_EXT_Set(uint32_t SetValue)
功能描述	设置 DeepSleep 模式下 RAM 数据保持
输入参数 1	DeepSleep 模式下 RAM 数据保持设置： 0：DeepSleep 模式下保持所有 RAM 数据 1：DeepSleep 模式下保持 16KB RAM 数据
输出参数	无
返回值	无

#### 4.2.14 PMU\_LPMC\_CFG\_DSLPRAM\_EXT\_Get

函数名	PMU_LPMC_CFG_DSLPRAM_EXT_Get
函数原型	uint32_t PMU_LPMC_CFG_DSLPRAM_EXT_Get(void)
功能描述	获取 DeepSleep 模式下 RAM 数据保持设置
输入参数 1	无
输出参数	无
返回值	DeepSleep 模式下 RAM 数据保持设置： 0：DeepSleep 模式下保持所有 RAM 数据 1：DeepSleep 模式下保持 16KB RAM 数据

#### 4.2.15 PMU\_LPREIE\_SLPEIE\_Setable

函数名	PMU_LPREIE_SLPEIE_Setable
函数原型	void PMU_LPREIE_SLPEIE_Setable(FunState NewState)
功能描述	设置 Sleep 错误中断使能
输入参数 1	Sleep 错误中断使能状态： DISABLE：禁止 Sleep 错误中断 ENABLE：使能 Sleep 错误中断
输出参数	无



返回值	无
-----	---

#### 4.2.16 PMU\_LPREIE\_SLPEIE\_Getable

函数名	PMU_LPREIE_SLPEIE_Getable
函数原型	FunState PMU_LPREIE_SLPEIE_Getable(void)
功能描述	获取 Sleep 错误中断使能状态
输入参数 1	无
输出参数	无
返回值	Sleep 错误中断使能状态： DISABLE：禁止 LSleep 错误中断 ENABLE：使能 Sleep 错误中断

#### 4.2.17 PMU\_LPREIF\_SLPEIF\_Clr

函数名	PMU_LPREIF_SLPEIF_Clr
函数原型	void PMU_LPREIF_SLPEIF_Clr(void)
功能描述	清除 Sleep 错误中断标志
输入参数 1	Sleep 错误中断使能标志： 0：未发生 Sleep 错误中断 1：发生 Sleep 错误中断
输出参数	无
返回值	无

#### 4.2.18 PMU\_LPREIF\_SLPEIF\_Chk

函数名	PMU_LPREIF_SLPEIF_Chk
函数原型	FlagStatus PMU_LPREIF_SLPEIF_Chk(void)
功能描述	获取 Sleep 错误中断标志
输入参数 1	无
输出参数	无
返回值	Sleep 错误中断使能标志： 0：未发生 Sleep 错误中断 1：发生 Sleep 错误中断

#### 4.2.19 PMU\_WKDLYCON\_TRHR\_Set

函数名	PMU_WKDLYCON_TRHR_Set
-----	-----------------------



函数原型	void PMU_WKDLYCON_TRHR_Set(uint32_t SetValue)
功能描述	设置 BG 使能到开启 LDO 并开始读取 Flash 的时间
输入参数 1	DeepSleep 唤醒时,从 BG 使能到开启 LDO 并开始读取 Flash 的时间间隔: 00: 12us 01: 16us 10: 24us 11: 32us
输出参数	无
返回值	无

#### 4.2.20 PMU\_WKDLYCON\_TRHR\_Get

函数名	PMU_WKDLYCON_TRHR_Get
函数原型	uint32_t PMU_WKDLYCON_TRHR_Get(void)
功能描述	获取 BG 使能到开启 LDO 并开始读取 Flash 的时间设置
输入参数 1	无
输出参数	无
返回值	DeepSleep 唤醒时,从 BG 使能到开启 LDO 并开始读取 Flash 的时间间隔: 00: 12us 01: 16us 10: 24us 11: 32us

#### 4.2.21 PMU\_WKPFLAG\_DBGWKF\_Clr

函数名	PMU_WKPFLAG_DBGWKF_Clr
函数原型	void PMU_WKPFLAG_DBGWKF_Clr(void)
功能描述	清除 CPU Debugger 唤醒标志
输入参数 1	无
输出参数	无
返回值	无

#### 4.2.22 PMU\_WKPFLAG\_DBGWKF\_Chk

函数名	PMU_WKPFLAG_DBGWKF_Chk
函数原型	FlagStatus PMU_WKPFLAG_DBGWKF_Chk(void)
功能描述	获取 CPU Debugger 唤醒标志



输入参数 1	无
输出参数	无
返回值	CPU Debugger 唤醒标志： 0: CPU Debugger 未唤醒 1: CPU Debugger 唤醒

#### 4.2.23 PMU\_WKPFLAG\_WKPxF\_ClrEx

函数名	PMU_WKPFLAG_WKPxF_ClrEx
函数原型	void PMU_WKPFLAG_WKPxF_ClrEx(uint32_t NWKPinDef)
功能描述	清除 NWKUPx Pin 唤醒标志
输入参数 1	NWKUPx Pin 唤醒标志： 0: 不清除 NWKUPx Pin 唤醒标志 1: 清除 NWKUPx Pin 唤醒标志
输出参数	无
返回值	无

#### 4.2.24 PMU\_WKPFLAG\_WKPxF\_ChkEx

函数名	PMU_WKPFLAG_WKPxF_ChkEx
函数原型	FlagStatus PMU_WKPFLAG_WKPxF_ChkEx(uint32_t NWKPinDef)
功能描述	获取 NWKUPx Pin 唤醒标志
输入参数 1	无
输出参数	无
返回值	NWKUPx Pin 唤醒标志： 0: 不清除 NWKUPx Pin 唤醒标志 1: 清除 NWKUPx Pin 唤醒标志

#### 4.2.25 PMU\_LPMCFG\_XTOFF\_Set

函数名	PMU_LPMCFG_XTOFF_Set
函数原型	void PMU_LPMCFG_XTOFF_Set(uint32_t SetValue)
功能描述	设置 XTTF 使能
输入参数 1	禁止 XTTF, 仅在 Sleep/DeepSleep 下起作用： 0: 保持 XTTF 开启 1: 禁止 XTTF
输出参数	无
返回值	无

#### 4.2.26 PMU\_LPMCFG\_XTOFF\_Get

函数名	PMU_LPMCFG_XTOFF_Get
函数原型	uint32_t PMU_LPMCFG_XTOFF_Get(void)
功能描述	获取 XTTF 使能状态
输入参数 1	无
输出参数	无
返回值	禁止 XTTF，仅在 Sleep/DeepSleep 下起作用： 0：保持 XTTF 开启 1：禁止 XTTF

#### 4.2.27 PMU\_LPMCFG\_LDO15EN\_Chk

函数名	PMU_LPMCFG_LDO15EN_Chk
函数原型	FlagStatus PMU_LPMCFG_LDO15EN_Chk(void)
功能描述	获取 LDO15 使能状态
输入参数 1	无
输出参数	无
返回值	LDO15 使能状态： DISABLE：禁止 LDO15 ENABLE：使能 LDO15

## 5 FLASH

### 5.1 FLASH 寄存器结构

寄存器	描述
FLSRDCON	Flash 读取控制寄存器
PRFTCON	预取指控制寄存器
OPTBR	用户配置寄存器
ACLOCK1	代码锁定寄存器 1
ACLOCK2	代码锁定寄存器 2
EPCON	Flash 擦写控制寄存器
FLSKEY	Flash Key 输入寄存器
FLSIE	Flash 中断使能寄存器
FLSIF	Flash 中断标志寄存器

### 5.2 FLASH 函数库

序号	函数名	描述
1	FLASH_Deinit	FLASH 设置初始化为缺省值
2	FLASH_FLSRDCON_WAIT_Set	设置 FLASH 读等待周期
3	FLASH_FLSRDCON_WAIT_Get	获取 FLASH 读等待周期设置
4	FLASH_Prog_SingleByte	单字节写 FLASH
5	FLASH_Prog_ByteString	连续写 FLASH
6	FLASH_Erase_Sector	块擦设置
7	FLASH_OPTBR_DBGCFGEN_Chk	获取 DBG 控制寄存器使能状态
8	FLASH_OPTBR_RAMBOOT_Chk	获取 RAM 引导使能状态
9	FLASH_OPTBR_ACLOCKEN_Chk	获取应用代码权限锁定使能状态
10	FLASH_OPTBR_DBRDPEN_Chk	获取 Debug Port 读取保护使能状态
11	FLASH_EPCON_ERTYPE_Set	设置 FLASH 扇区擦除方式
12	FLASH_EPCON_ERTYPE_Get	获取 FLASH 扇区擦除方式
13	FLASH_EPCON_PREQ_Set	设置编程请求
14	FLASH_EPCON_PREQ_Get	获取编程请求状态
15	FLASH_EPCON_EREQ_Set	设置擦除请求



16	FLASH_EPCON_EREQ_Get	获取擦除请求状态
17	FLASH_FLSIE_AUTHIE_Setable	设置 FLASH 擦写 key 输入使能
18	FLASH_FLSIE_AUTHIE_Getable	获取 FLASH 擦写 key 输入使能状态
19	FLASH_FLSIE_KEYIE_Setable	设置 FLASH key 错误中断使能
20	FLASH_FLSIE_KEYIE_Getable	获取 FLASH key 错误中断使能状态
21	FLASH_FLSIE_CKIE_Setable	设置擦写定时时钟错误中断使能
22	FLASH_FLSIE_CKIE_Getable	获取擦写定时时钟错误中断使能状态
23	FLASH_FLSIE_PRDIE_Setable	设置编程完成标志中断使能
24	FLASH_FLSIE_PRDIE_Getable	获取编程完成标志中断使能状态
25	FLASH_FLSIE_ERDIE_Setable	设置擦写完成标志中断使能
26	FLASH_FLSIE_ERDIE_Getable	获取擦写完成标志中断使能状态
27	FLASH_FLSIF_AUTHIF_Clr	对 LOCK 块数据清零
28	FLASH_FLSIF_AUTHIF_Chk	获取 FLASH 读写权限错误
29	FLASH_FLSIF_KEYIF_Clr	清除 FLASH key 错误
30	FLASH_FLSIF_KEYIF_Chk	获取 FLASH key 错误状态
31	FLASH_FLSIF_CKIF_Clr	清除 FLASH CK 错误
32	FLASH_FLSIF_CKIF_Chk	获取 FLASH CK 错误状态
33	FLASH_FLSIF_PRDIF_Clr	清除编程完成标志
34	FLASH_FLSIF_PRDIF_Chk	获取编程完成标志状态
35	FLASH_FLSIF_ERDIF_Clr	清除擦写完成标志
36	FLASH_FLSIF_ERDIF_Chk	获取擦写完成标志状态
37	FLASH_FLSKEY_Write	输入合法 Key 序列,无物理寄存器
38	FLASH_ACLOCK1_Write	控制 Block31~Block0 的应用代码读写锁定
39	FLASH_ACLOCK1_Read	获取 Block31~Block0 的应用代码读写锁定状态
40	FLASH_ACLOCK2_Write	控制 Block63~Block32 的应用代码读写锁定
41	FLASH_ACLOCK2_Read	获取 Block63~Block32 的应用代码读写锁定状态

### 5.2.1 FLASH\_Deinit

函数名	FLASH_Deinit
函数原型	void FLASH_Deinit(void)
功能描述	FLASH 设置初始化为缺省值
输入参数 1	无
输出参数	无
返回值	无

### 5.2.2 FLASH\_FLSRDCON\_WAIT\_Set

函数名	FLASH_FLSRDCON_WAIT_Set
函数原型	void FLASH_FLSRDCON_WAIT_Set(uint32_t SetValue)
功能描述	设置 FLASH 读等待周期
输入参数 1	FLASH 读等待周期设置: 00/11: 0 wait cycle 01: 1wait cycle 10: 2 wait cycle
输出参数	无
返回值	无

### 5.2.3 FLASH\_FLSRDCON\_WAIT\_Get

函数名	FLASH_FLSRDCON_WAIT_Get
函数原型	uint32_t FLASH_FLSRDCON_WAIT_Get(void)
功能描述	获取 FLASH 读等待周期设置
输入参数 1	无
输出参数	无
返回值	FLASH 读等待周期设置: 00/11: 0 wait cycle 01: 1wait cycle 10: 2 wait cycle

### 5.2.4 FLASH\_Prog\_SingleByte

函数名	FLASH_Prog_SingleByte
函数原型	void FLASH_Prog_SingleByte(uint32_t prog_addr,uint8_t prog_data)



功能描述	单字节写 FLASH
输入参数 1	写入的地址
输入参数 2	写入的数据
输出参数	无
返回值	无

### 5.2.5 FLASH\_Prog\_ByteString

函数名	FLASH_Prog_ByteString
函数原型	void FLASH_Prog_ByteString(uint32_t prog_addr, uint8_t* prog_data, uint16_t Len)
功能描述	连续写 FLASH
输入参数 1	写入的地址
输入参数 2	写入的数据
输入参数 3	数据长度
输出参数	无
返回值	无

### 5.2.6 FLASH\_Erase\_Sector

函数名	FLASH_Erase_Sector
函数原型	void FLASH_Erase_Sector(uint32_t erase_addr)
功能描述	块擦设置
输入参数 1	擦除地址
输出参数	无
返回值	无

### 5.2.7 FLASH\_OPTBR\_DBGCFGEN\_Chk

函数名	FLASH_OPTBR_DBGCFGEN_Chk
函数原型	FlagStatus FLASH_OPTBR_DBGCFGEN_Chk(void)
功能描述	获取 DBG 控制寄存器使能状态
输入参数 1	无
输出参数	无
返回值	DBG 控制寄存器使能状态： DISABLE：关闭 MCUDBGCR 寄存器功能 ENABLE：使能 MCUDBGCR 寄存器功能

### 5.2.8 FLASH\_OPTBR\_RAMBOOT\_Chk

函数名	FLASH_OPTBR_RAMBOOT_Chk
函数原型	FlagStatus FLASH_OPTBR_RAMBOOT_Chk(void)
功能描述	获取 RAM 引导使能状态
输入参数 1	无
输出参数	无
返回值	RAM 引导使能状态： DISABLE： 禁止芯片启动时从 RAM 引导 ENABLE： 允许芯片启动时从 RAM 引导

### 5.2.9 FLASH\_OPTBR\_ACLOCKEN\_Chk

函数名	FLASH_OPTBR_ACLOCKEN_Chk
函数原型	FlagStatus FLASH_OPTBR_ACLOCKEN_Chk(void)
功能描述	获取应用代码权限锁定使能状态
输入参数 1	无
输出参数	无
返回值	应用代码权限锁定使能状态： 00/01/11： 禁止 ACLOCK 10： 使能 ACLOCK

### 5.2.10 FLASH\_OPTBR\_DBRDPEN\_Chk

函数名	FLASH_OPTBR_DBRDPEN_Chk
函数原型	FlagStatus FLASH_OPTBR_DBRDPEN_Chk(void)
功能描述	获取 Debug Port 读取保护使能状态
输入参数 1	无
输出参数	无
返回值	Debug Port 读取保护使能状态： 00/01/11： 禁止 DBRDP 10： 使能 DBRDP

### 5.2.11 FLASH\_EPCON\_ERTYPE\_Set

函数名	FLASH_EPCON_ERTYPE_Set
函数原型	void FLASH_EPCON_ERTYPE_Set(uint32_t SetValue)
功能描述	设置 FLASH 扇区擦除方式

输入参数 1	FLASH 扇区擦除方式： 00/11: Sector Erase 扇擦除 01: Block Erase 块擦除 1DISABLE: 禁止
输出参数	无
返回值	无

### 5.2.12 FLASH\_EPCON\_ERTYPE\_Get

函数名	FLASH_EPCON_ERTYPE_Get
函数原型	uint32_t FLASH_EPCON_ERTYPE_Get(void)
功能描述	获取 FLASH 扇区擦除方式
输入参数 1	无
输出参数	无
返回值	FLASH 扇区擦除方式： 00/11: Sector Erase 扇擦除 01: Block Erase 块擦除 1DISABLE: 禁止

### 5.2.13 FLASH\_EPCON\_PREQ\_Set

函数名	FLASH_EPCON_PREQ_Set
函数原型	void FLASH_EPCON_PREQ_Set(uint32_t SetValue)
功能描述	设置编程请求
输入参数 1	编程请求： 0: 不置位 EPCON 的 PREQ 1: 置位 EPCON 的 PREQ,硬件完成编程后自动清零
输出参数	无
返回值	无

### 5.2.14 FLASH\_EPCON\_PREQ\_Get

函数名	FLASH_EPCON_PREQ_Get
函数原型	uint32_t FLASH_EPCON_PREQ_Get(void)
功能描述	获取编程请求状态
输入参数 1	无
输出参数	无
返回值	编程请求：

	0: 不置位 EPCON 的 PREQ 1: 置位 EPCON 的 PREQ,硬件完成编程后自动清零
--	-------------------------------------------------------

### 5.2.15 FLASH\_EPCON\_EREQ\_Set

函数名	FLASH_EPCON_EREQ_Set
函数原型	void FLASH_EPCON_EREQ_Set(uint32_t SetValue)
功能描述	设置擦除请求
输入参数 1	擦除请求: 0: 不置位 EPCON 的 EREQ 1: 置位 EPCON 的 EREQ,硬件完成编程后自动清零
输出参数	无
返回值	无

### 5.2.16 FLASH\_EPCON\_EREQ\_Get

函数名	FLASH_EPCON_EREQ_Get
函数原型	uint32_t FLASH_EPCON_EREQ_Get(void)
功能描述	获取擦除请求状态
输入参数 1	无
输出参数	无
返回值	擦除请求: 0: 不置位 EPCON 的 EREQ 1: 置位 EPCON 的 EREQ,硬件完成编程后自动清零

### 5.2.17 FLASH\_FLSIE\_AUTHIE\_Setable

函数名	FLASH_FLSIE_AUTHIE_Setable
函数原型	void FLASH_FLSIE_AUTHIE_Setable(FunState NewState)
功能描述	设置 FLASH 擦写 key 输入使能
输入参数 1	FLASH 擦写 key 输入使能: DISABLE: 禁止 FLASH 擦写 key 输入 ENABLE: 使能 FLASH 擦写 key 输入
输出参数	无
返回值	无

### 5.2.18 FLASH\_FLSIE\_AUTHIE\_Getable

函数名	FLASH_FLSIE_AUTHIE_Getable
函数原型	FunState FLASH_FLSIE_AUTHIE_Getable(void)
功能描述	获取 FLASH 擦写 key 输入使能状态
输入参数 1	无
输出参数	无
返回值	FLASH 擦写 key 输入使能： DISABLE: 禁止 FLASH 擦写 key 输入 ENABLE: 使能 FLASH 擦写 key 输入

### 5.2.19 FLASH\_FLSIE\_KEYIE\_Setable

函数名	FLASH_FLSIE_KEYIE_Setable
函数原型	void FLASH_FLSIE_KEYIE_Setable(FunState NewState)
功能描述	设置 FLASH key 错误中断使能
输入参数 1	FLASH key 错误中断使能： DISABLE: 禁止 FLASH key 错误中断 ENABLE: 使能 FLASH key 错误中断
输出参数	无
返回值	无

### 5.2.20 FLASH\_FLSIE\_KEYIE\_Getable

函数名	FLASH_FLSIE_KEYIE_Getable
函数原型	FunState FLASH_FLSIE_KEYIE_Getable(void)
功能描述	获取 FLASH key 错误中断使能状态
输入参数 1	无
输出参数	无
返回值	FLASH key 错误中断使能： DISABLE: 禁止 FLASH key 错误中断 ENABLE: 使能 FLASH key 错误中断

### 5.2.21 FLASH\_FLSIE\_CKIE\_Setable

函数名	FLASH_FLSIE_CKIE_Setable
函数原型	void FLASH_FLSIE_CKIE_Setable(FunState NewState)
功能描述	设置擦写定时时钟错误中断使能

输入参数 1	擦写定时时钟错误中断使能： DISABLE：禁止擦写定时时钟错误中断 ENABLE：使能擦写定时时钟错误中断
输出参数	无
返回值	无

### 5.2.22 FLASH\_FLSIE\_CKIE\_Getable

函数名	FLASH_FLSIE_CKIE_Getable
函数原型	FunState FLASH_FLSIE_CKIE_Getable(void)
功能描述	清获取擦写定时时钟错误中断使能状态
输入参数 1	无
输出参数	无
返回值	擦写定时时钟错误中断使能： DISABLE：禁止擦写定时时钟错误中断 ENABLE：使能擦写定时时钟错误中断

### 5.2.23 FLASH\_FLSIE\_PRDIE\_Setable

函数名	FLASH_FLSIE_PRDIE_Setable
函数原型	void FLASH_FLSIE_PRDIE_Setable(FunState NewState)
功能描述	设置编程完成标志中断使能
输入参数 1	编程完成标志中断使能： DISABLE：禁止编程完成标志中断 ENABLE：使能编程完成标志中断
输出参数	无
返回值	无

### 5.2.24 FLASH\_FLSIE\_PRDIE\_Getable

函数名	FLASH_FLSIE_PRDIE_Getable
函数原型	FunState FLASH_FLSIE_PRDIE_Getable(void)
功能描述	获取编程完成标志中断使能状态
输入参数 1	无
输出参数	无
返回值	编程完成标志中断使能： DISABLE：禁止编程完成标志中断 ENABLE：使能编程完成标志中断

### 5.2.25 FLASH\_FLSIE\_ERDIE\_Setable

函数名	FLASH_FLSIE_ERDIE_Setable
函数原型	void FLASH_FLSIE_ERDIE_Setable(FunState NewState)
功能描述	设置擦写完成标志中断使能
输入参数 1	擦写完成标志中断使能： DISABLE：禁止擦写完成标志中断 ENABLE：使能擦写完成标志中断
输出参数	无
返回值	无

### 5.2.26 FLASH\_FLSIE\_ERDIE\_Getable

函数名	FLASH_FLSIE_ERDIE_Getable
函数原型	FunState FLASH_FLSIE_ERDIE_Getable(void)
功能描述	获取擦写完成标志中断使能状态
输入参数 1	无
输出参数	无
返回值	擦写完成标志中断使能： DISABLE：禁止擦写完成标志中断 ENABLE：使能擦写完成标志中断

### 5.2.27 FLASH\_FLSIF\_AUTHIF\_Clr

函数名	FLASH_FLSIF_AUTHIF_Clr
函数原型	void FLASH_FLSIF_AUTHIF_Clr(void)
功能描述	对 LOCK 块数据清零
输入参数 1	无
输出参数	无
返回值	无

### 5.2.28 FLASH\_FLSIF\_AUTHIF\_Chk

函数名	FLASH_FLSIF_AUTHIF_Chk
函数原型	FlagStatus FLASH_FLSIF_AUTHIF_Chk(void)
功能描述	获取 FLASH 读写权限错误
输入参数 1	无

输出参数	无
返回值	FLASH 读写权限错误： 0：未产生 FLASH 读写权限错误中断 1：产生 FLASH 读写权限错误中断

### 5.2.29 FLASH\_FLSIF\_KEYIF\_Clr

函数名	FLASH_FLSIF_KEYIF_Clr
函数原型	void FLASH_FLSIF_KEYIF_Clr(void)
功能描述	清除 FLASH key 错误
输入参数 1	无
输出参数	无
返回值	无

### 5.2.30 FLASH\_FLSIF\_KEYIF\_Chk

函数名	FLASH_FLSIF_KEYIF_Chk
函数原型	FlagStatus FLASH_FLSIF_KEYIF_Chk(void)
功能描述	清除 FLASH key 错误
输入参数 1	无
输出参数	无
返回值	清除 FLASH key 错误： 0：未产生 FLASH key 错误中断 1：产生 FLASH key 错误中断

### 5.2.31 FLASH\_FLSIF\_CKIF\_Clr

函数名	FLASH_FLSIF_CKIF_Clr
函数原型	void FLASH_FLSIF_CKIF_Clr(void)
功能描述	清除 FLASH 擦写定时时钟错误
输入参数 1	无
输出参数	无
返回值	无

### 5.2.32 FLASH\_FLSIF\_CKIF\_Chk

函数名	FLASH_FLSIF_CKIF_Chk
函数原型	FlagStatus FLASH_FLSIF_CKIF_Chk(void)



功能描述	获取 FLASH 擦写定时时钟错误状态
输入参数 1	无
输出参数	无
返回值	FLASH 擦写定时时钟错误状态： 0：未产生 FLASH 擦写定时时钟错误中断 1：产生 FLASH 擦写定时时钟错误中断

### 5.2.33 FLASH\_FLSIF\_PRDIF\_Clr

函数名	FLASH_FLSIF_PRDIF_Clr
函数原型	void FLASH_FLSIF_PRDIF_Clr(void)
功能描述	清除编程完成标志
输入参数 1	无
输出参数	无
返回值	无

### 5.2.34 FLASH\_FLSIF\_PRDIF\_Chk

函数名	FLASH_FLSIF_PRDIF_Chk
函数原型	FlagStatus FLASH_FLSIF_PRDIF_Chk(void)
功能描述	获取编程完成标志状态
输入参数 1	无
输出参数	无
返回值	编程完成标志状态： 0：未产生编程完成中断 1：产生编程完成中断

### 5.2.35 FLASH\_FLSIF\_ERDIF\_Clr

函数名	FLASH_FLSIF_ERDIF_Clr
函数原型	void FLASH_FLSIF_ERDIF_Clr(void)
功能描述	清除擦写完成标志
输入参数 1	无
输出参数	无
返回值	无

### 5.2.36 FLASH\_FLSIF\_ERDIF\_Chk

函数名	FLASH_FLSIF_ERDIF_Chk
函数原型	FlagStatus FLASH_FLSIF_ERDIF_Chk(void)
功能描述	获取擦写完成标志状态
输入参数 1	无
输出参数	无
返回值	擦写完成标志状态： 0: 未产生擦写完成中断 1: 产生擦写完成中断

### 5.2.37 FLASH\_FLSKEY\_Write

函数名	FLASH_FLSKEY_Write
函数原型	void FLASH_FLSKEY_Write(uint32_t SetValue)
功能描述	输入合法 Key 序列,无物理寄存器
输入参数 1	合法 Key 序列
输出参数	无
返回值	无

### 5.2.38 FLASH\_ACLOCK1\_Write

函数名	FLASH_ACLOCK1_Write
函数原型	void FLASH_ACLOCK1_Write(uint32_t SetValue)
功能描述	控制 Block31~Block0 的应用代码读写锁定
输入参数 1	Block31~Block0 的每一位应用代码读写锁定： 对应的任意位为 0: 开启读取和擦除权限 对应的任意位为 1: 锁定读取和擦除权限
输出参数	无
返回值	无

### 5.2.39 FLASH\_ACLOCK1\_Read

函数名	FLASH_ACLOCK1_Read
函数原型	uint32_t FLASH_ACLOCK1_Read(void)
功能描述	获取 Block31~Block0 的应用代码读写锁定状态
输入参数 1	无
输出参数	无

返回值	Block31~Block0 的每一位应用代码读写锁定： 对应的任意位为 0：开启读取和擦除权限 对应的任意位为 1：锁定读取和擦除权限
-----	----------------------------------------------------------------------------

#### 5.2.40 FLASH\_ACLOCK2\_Write

函数名	FLASH_ACLOCK2_Write
函数原型	void PMU_WKPFLAG_DBGWKF_Clr(void)
功能描述	控制 Block63~Block32 的应用代码读写锁定
输入参数 1	Block63~Block32 的每一位应用代码读写锁定： 对应的任意位为 0：开启读取和擦除权限 对应的任意位为 1：锁定读取和擦除权限
输出参数	无
返回值	无

#### 5.2.41 FLASH\_ACLOCK2\_Read

函数名	FLASH_ACLOCK2_Read
函数原型	uint32_t FLASH_ACLOCK2_Read(void)
功能描述	获取 Block63~Block32 的应用代码读写锁定状态
输入参数 1	无
输出参数	无
返回值	Block63~Block32 的每一位应用代码读写锁定： 对应的任意位为 0：开启读取和擦除权限 对应的任意位为 1：锁定读取和擦除权限

## 6 IWDT

### 6.1 IWDT 寄存器结构

寄存器	描述
IWDTSERV	IWDT 清除寄存器
IWDTCFG	IWDT 设置寄存器
IWDTCNT	IWDT 计数值寄存器

### 6.2 IWDT 函数库

序号	函数名	描述
1	IWDT_Deinit	IWDT 初始化为缺省值
2	IWDT_Clr	清除 IWDT
3	IWDT_IWDTSERV_Write	写入 0x1234_5A5A 清除 IWDT
4	IWDT_IWDTCNT_Read	读取 IWDT 当前计数值
5	IWDT_IWDTCFG_IWDTOVP_Set	设置 IWDT 溢出周期
6	IWDT_IWDTCFG_IWDTOVP_Get	获取 IWDT 溢出周期
7	IWDT_IWDTCFG_IWDTSLP4096S_Setable	设置 IWDT 溢出周期为 4096s
8	IWDT_IWDTCFG_IWDTSLP4096S_Getable	获取 IWDT 溢出周期是否为 4096s

#### 6.2.1 IWDT\_Deinit

函数名	IWDT_Deinit
函数原型	void IWDT_Deinit(void)
功能描述	IWDT 初始化为缺省值
输入参数 1	无
输出参数	无
返回值	无

#### 6.2.2 IWDT\_Clr

函数名	IWDT_Clr
函数原型	void IWDT_Clr(void)
功能描述	清除 IWDT

输入参数 1	无
输出参数	无
返回值	无

### 6.2.3 IWDI\_IWDTSERV\_Write

函数名	IWDI_IWDTSERV_Write
函数原型	void IWDI_IWDTSERV_Write(uint32_t SetValue)
功能描述	写入 0x1234_5A5A 清除 IWDI
输入参数 1	写入 0x1234_5A5A 清除 IWDI
输出参数	无
返回值	无

### 6.2.4 IWDI\_IWDTCNT\_Read

函数名	IWDI_IWDTCNT_Read
函数原型	uint32_t IWDI_IWDTCNT_Read(void)
功能描述	读取 IWDI 当前计数值
输入参数 1	无
输出参数	无
返回值	IWDI 当前计数值: 0-3FFFF

### 6.2.5 IWDI\_IWDTCFG\_IWDTOVP\_Set

函数名	IWDI_IWDTCFG_IWDTOVP_Set
函数原型	void IWDI_IWDTCFG_IWDTOVP_Set(uint32_t SetValue)
功能描述	设置 IWDI 溢出周期
输入参数 1	IWDI 溢出周期: 0X00: 125ms 0X01: 500ms 0X10: 2s 0X11: 8s
输出参数	无
返回值	无

### 6.2.6 IWDI\_IWDTCFG\_IWDTOVP\_Get

函数名	IWDI_IWDTCFG_IWDTOVP_Get
-----	--------------------------

函数原型	uint32_t IWDT_IWDTCFG_IWDTOVP_Get(void)
功能描述	获取 IWDT 溢出周期
输入参数 1	无
输出参数	无
返回值	IWDT 溢出周期： 0X00: 125ms 0X01: 500ms 0X10: 2s 0X11: 8s

### 6.2.7 IWDT\_IWDTCFG\_IWDTSLP4096S\_Setable

函数名	IWDT_IWDTCFG_IWDTSLP4096S_Setable
函数原型	void IWDT_IWDTCFG_IWDTSLP4096S_Setable(FunState NewState)
功能描述	是否设置 IWDT 溢出周期为 4096s： 0: 不设置 IWDT 溢出周期为 4096s 1: 设置 IWDT 溢出周期为 4096s
输入参数 1	无
输出参数	无
返回值	无

### 6.2.8 IWDT\_IWDTCFG\_IWDTSLP4096S\_Getable

函数名	IWDT_IWDTCFG_IWDTSLP4096S_Getable
函数原型	FunState IWDT_IWDTCFG_IWDTSLP4096S_Getable(void)
功能描述	获取 IWDT 溢出周期是否为 4096s
输入参数 1	无
输出参数	无
返回值	是否设置 IWDT 溢出周期为 4096s： 0: 不设置 IWDT 溢出周期为 4096s 1: 设置 IWDT 溢出周期为 4096s

## 7 WWDT

### 7.1 WWDT 寄存器结构

寄存器	描述
WWDTCON	WWDT 控制寄存器
WWDTCFG	WWDT 设置寄存器
WWDTCNT	WWDT 计数值寄存器
WWDTIE	WWDT 中断使能寄存器
WWDTIF	WWDT 中断标志寄存器
WWDTDIV	WWDT 预分频寄存器

### 7.2 WWDT 函数库

序号	函数名	描述
1	WWDT_Deinit	WWDT 初始化为缺省值
2	WWDT_Start	启动 WWDT
3	WWDT_Clr	清除 WWDT
4	WWDT_WWDTCON_Write	写入 0xAC 启动 WWDT
5	WWDT_WWDTCNT_Read	读取 WWDT 当前计数值
6	WWDT_WWDTCFG_WWDTCFG_Set	设置 WWDT 溢出时间
7	WWDT_WWDTCFG_WWDTCFG_Get	获取 WWDT 溢出时间
8	WWDT_WWDTDIV_Read	读取 WWDT 的 4096 预分频设置值
9	WWDT_WWDTIE_WWDTIE_Setable	设置 WWDT 中断使能
10	WWDT_WWDTIE_WWDTIE_Getable	获取 WWDT 中断使能状态
11	WWDT_WWDTIF_WWDTIF_Clr	清除 WWDT 75%计时中断标志
12	WWDT_WWDTIF_WWDTIF_Chk	读取 WWDT 75%计时中断标志

#### 7.2.1 WWDT\_Deinit

函数名	WWDT_Deinit
函数原型	void WWDT_Deinit(void)
功能描述	WWDT 初始化为缺省值
输入参数 1	无



输出参数	无
返回值	无

### 7.2.2 WWDT\_Start

函数名	WWDT_Start
函数原型	void WWDT_start(void)
功能描述	启动 WWDT
输入参数 1	无
输出参数	无
返回值	无

### 7.2.3 WWDT\_Clr

函数名	WWDT_Clr
函数原型	void WWDT_Clr(void)
功能描述	清除 WWDT
输入参数 1	无
输出参数	无
返回值	无

### 7.2.4 WWDT\_WWDTCNT\_Write

函数名	WWDT_WWDTCNT_Write
函数原型	void WWDT_WWDTCNT_Write(uint32_t SetValue)
功能描述	写入 0xAC 启动 WWDT
输入参数 1	写入 0xAC 启动 WWDT
输出参数	无
返回值	无

### 7.2.5 WWDT\_WWDTCNT\_Read

函数名	WWDT_WWDTCNT_Read
函数原型	uint32_t WWDT_WWDTCNT_Read(void)
功能描述	读取 WWDT 当前计数值
输入参数 1	无
输出参数	无
返回值	WWDT 当前计数值:0-3FF



### 7.2.6 WWDT\_WWDTCFG\_WWDTCFG\_Set

函数名	WWDT_WWDTCFG_WWDTCFG_Set
函数原型	void WWDT_WWDTCFG_WWDTCFG_Set(uint32_t SetValue)
功能描述	设置 WWDT 溢出时间
输入参数 1	WWDT 溢出时间: 000: TPCLK * 4096 * 1 001: TPCLK * 4096 * 4 010: TPCLK * 4096 * 16 011: TPCLK * 4096 * 64 100: TPCLK * 4096 * 128 101: TPCLK * 4096 * 256 110: TPCLK * 4096 * 512 111: TPCLK * 4096 * 1024
输出参数	无
返回值	无

### 7.2.7 WWDT\_WWDTCFG\_WWDTCFG\_Get

函数名	WWDT_WWDTCFG_WWDTCFG_Get
函数原型	uint32_t WWDT_WWDTCFG_WWDTCFG_Get(void)
功能描述	获取 WWDT 溢出时间
输入参数 1	无
输出参数	无
返回值	WWDT 溢出时间: 000: TPCLK * 4096 * 1 001: TPCLK * 4096 * 4 010: TPCLK * 4096 * 16 011: TPCLK * 4096 * 64 100: TPCLK * 4096 * 128 101: TPCLK * 4096 * 256 110: TPCLK * 4096 * 512 111: TPCLK * 4096 * 1024

### 7.2.8 WWDT\_WWDTDIV\_Read

函数名	WWDT_WWDTDIV_Read
-----	-------------------

函数原型	uint32_t WWDT_WWDTDIV_Read(void)
功能描述	读取 WWDT 的 4096 预分频设置值
输入参数 1	无
输出参数	无
返回值	WWDT 的 4096 预分频设置值:0-FFF

### 7.2.9 WWDT\_WWDTIE\_WWDTIE\_Setable

函数名	WWDT_WWDTIE_WWDTIE_Setable
函数原型	void WWDT_WWDTIE_WWDTIE_Setable(FunState NewState)
功能描述	设置 WWDT 中断使能
输入参数 1	WWDT 中断使能状态: DISABLE: 禁止 WWDT 中断 ENABLE: 使能 WWDT 中断
输出参数	无
返回值	无

### 7.2.10 WWDT\_WWDTIE\_WWDTIE\_Getable

函数名	WWDT_WWDTIE_WWDTIE_Getable
函数原型	FunState WWDT_WWDTIE_WWDTIE_Getable(void)
功能描述	获取 WWDT 中断使能状态
输入参数 1	无
输出参数	无
返回值	WWDT 中断使能状态: DISABLE: 禁止 WWDT 中断 ENABLE: 使能 WWDT 中断

### 7.2.11 WWDT\_WWDTIF\_WWDTIF\_Clr

函数名	WWDT_WWDTIF_WWDTIF_Clr
函数原型	void WWDT_WWDTIF_WWDTIF_Clr(void)
功能描述	清除 WWDT 75%计时中断标志
输入参数 1	无
输出参数	无
返回值	无



## 7.2.12 WWDT\_WWDTIF\_WWDTIF\_Chk

函数名	WWDT_WWDTIF_WWDTIF_Chk
函数原型	FlagStatus WWDT_WWDTIF_WWDTIF_Chk(void)
功能描述	读取 WWDT 75%计时中断标志
输入参数 1	无
输出参数	无
返回值	WWDT 75%计时中断标志： 0: 未发生 WWDT 75%计时中断标志 1: 发生 WWDT 75%计时中断标志

## 8 SCU

### 8.1 SCU 寄存器结构

寄存器	描述
MCUDBGCR	MCU Debug 设置寄存器
HDFFLAG	HardFault 查询寄存器

### 8.2 SCU 函数库

序号	函数名	描述
1	SCU_Deinit	SCU 初始化为缺省值
2	SCU_MCUDBGCR_DBG_ET1_STOP_Setable	Debug 下设置 ET1 使能
3	SCU_MCUDBGCR_DBG_ET1_STOP_Getable	获取 Debug 下 ET1 使能状态
4	SCU_MCUDBGCR_DBG_ET2_STOP_Setable	Debug 下设置 ET2 使能
5	SCU_MCUDBGCR_DBG_ET2_STOP_Getable	获取 Debug 下 ET2 使能状态
6	SCU_MCUDBGCR_DBG_ET3_STOP_Setable	Debug 下设置 ET3 使能
7	SCU_MCUDBGCR_DBG_ET3_STOP_Getable	获取 Debug 下 ET3 使能状态
8	SCU_MCUDBGCR_DBG_ET4_STOP_Setable	Debug 下设置 ET4 使能
9	SCU_MCUDBGCR_DBG_ET4_STOP_Getable	获取 Debug 下 ET4 使能状态
10	SCU_MCUDBGCR_DBG_BT1_STOP_Setable	Debug 下设置 BT1 使能
11	SCU_MCUDBGCR_DBG_BT1_STOP_Getable	获取 Debug 下 BT1 使能状态
12	SCU_MCUDBGCR_DBG_BT2_STOP_Setable	Debug 下设置 BT2 使能
13	SCU_MCUDBGCR_DBG_BT2_STOP_Getable	获取 Debug 下 BT2 使能状态
14	SCU_MCUDBGCR_DBG_WWDT_STOP_Setable	Debug 下设置 WWDT 使能
15	SCU_MCUDBGCR_DBG_WWDT_STOP_Getable	获取 Debug 下 WWDT 使能状态
16	SCU_MCUDBGCR_DBG_IWDT_STOP_Setable	Debug 下设置 IWDT 使能
17	SCU_MCUDBGCR_DBG_IWDT_STOP_Getable	获取 Debug 下 IWDT 使能状态
18	SCU_HDFFLAG_HDF_REQUEST_FLAG_Clr	清除 HardFault 标志位
19	SCU_HDFFLAG_HDF_REQUEST_FLAG_Chk	获取 HardFault 标志位
20	SCU_HDFFLAG_DABORT_ADDR_FLAG_Clr	清除地址非对齐访问错误标志
21	SCU_HDFFLAG_DABORT_ADDR_FLAG_Chk	获取地址非对齐访问错误标志
22	SCU_HDFFLAG_DABORT_RESP_FLAG_Clr	清除非法地址访问错误标志
23	SCU_HDFFLAG_DABORT_RESP_FLAG_Chk	获取非法地址访问错误标志

24	SCU_HDFFLAG_SVCUNDEF_FLAG_Clr	清除 SVC instructions 未定义标志
25	SCU_HDFFLAG_SVCUNDEF_FLAG_Chk	获取 SVC instructions 未定义标志
26	SCU_HDFFLAG_BKPT_FLAG_Clr	清除执行 BKPT 指令标志
27	SCU_HDFFLAG_BKPT_FLAG_Chk	获取执行 BKPT 指令标志
28	SCU_HDFFLAG_TBIT_FLAG_Clr	清除 Thumb-State 标志
29	SCU_HDFFLAG_TBIT_FLAG_Chk	获取 Thumb-State 标志
30	SCU_HDFFLAG_SPECIAL_OP_FLAG_Clr	清除特殊指令标志
31	SCU_HDFFLAG_SPECIAL_OP_FLAG_Chk	获取特殊指令标志

### 8.2.1 SCU\_Deinit

函数名	SCU_Deinit
函数原型	void SCU_Deinit(void)
功能描述	SCU 初始化为缺省值
输入参数 1	无
输出参数	无
返回值	无

### 8.2.2 SCU\_MCUDBGCR\_DBG\_ET1\_STOP\_Setable

函数名	SCU_MCUDBGCR_DBG_ET1_STOP_Setable
函数原型	void SCU_MCUDBGCR_DBG_ET1_STOP_Setable(FunState NewState)
功能描述	Debug 下设置 ET1 使能
输入参数 1	Debug 下 ET1 使能: DISABLE: 禁止 ET1 ENABLE: 使能 ET1
输出参数	无
返回值	无

### 8.2.3 SCU\_MCUDBGCR\_DBG\_ET1\_STOP\_Getable

函数名	SCU_MCUDBGCR_DBG_ET1_STOP_Getable
函数原型	FunState SCU_MCUDBGCR_DBG_ET1_STOP_Getable(void)
功能描述	获取 Debug 下 ET1 使能状态
输入参数 1	无
输出参数	无

返回值	Debug 下 ET1 使能: DISABLE: 禁止 ET1 ENABLE: 使能 ET1
-----	------------------------------------------------------

#### 8.2.4 SCU\_MCUDBGCR\_DBG\_ET2\_STOP\_Setable

函数名	SCU_MCUDBGCR_DBG_ET2_STOP_Setable
函数原型	void SCU_MCUDBGCR_DBG_ET2_STOP_Setable(FunState NewState)
功能描述	Debug 下设置 ET2 使能
输入参数 1	Debug 下 ET2 使能: DISABLE: 禁止 ET2 ENABLE: 使能 ET2
输出参数	无
返回值	无

#### 8.2.5 SCU\_MCUDBGCR\_DBG\_ET2\_STOP\_Getable

函数名	SCU_MCUDBGCR_DBG_ET2_STOP_Getable
函数原型	FunState SCU_MCUDBGCR_DBG_ET2_STOP_Getable(void)
功能描述	获取 Debug 下 ET2 使能状态
输入参数 1	无
输出参数	无
返回值	Debug 下 ET2 使能: DISABLE: 禁止 ET2 ENABLE: 使能 ET2

#### 8.2.6 SCU\_MCUDBGCR\_DBG\_ET3\_STOP\_Setable

函数名	SCU_MCUDBGCR_DBG_ET3_STOP_Setable
函数原型	void SCU_MCUDBGCR_DBG_ET3_STOP_Setable(FunState NewState)
功能描述	Debug 下设置 ET3 使能
输入参数 1	Debug 下 ET3 使能: DISABLE: 禁止 ET3 ENABLE: 使能 ET3
输出参数	无
返回值	无

### 8.2.7 SCU\_MCUDBGCR\_DBG\_ET3\_STOP\_Getable

函数名	SCU_MCUDBGCR_DBG_ET3_STOP_Getable
函数原型	FunState SCU_MCUDBGCR_DBG_ET3_STOP_Getable(void)
功能描述	获取 Debug 下 ET3 使能状态
输入参数 1	无
输出参数	无
返回值	Debug 下 ET3 使能: DISABLE: 禁止 ET3 ENABLE: 使能 ET3

### 8.2.8 SCU\_MCUDBGCR\_DBG\_ET4\_STOP\_Setable

函数名	SCU_MCUDBGCR_DBG_ET4_STOP_Setable
函数原型	void SCU_MCUDBGCR_DBG_ET4_STOP_Setable(FunState NewState)
功能描述	Debug 下设置 ET4 使能
输入参数 1	Debug 下 ET4 使能: DISABLE: 禁止 ET4 ENABLE: 使能 ET4
输出参数	无
返回值	无

### 8.2.9 SCU\_MCUDBGCR\_DBG\_ET4\_STOP\_Getable

函数名	SCU_MCUDBGCR_DBG_ET4_STOP_Getable
函数原型	FunState SCU_MCUDBGCR_DBG_ET4_STOP_Getable(void)
功能描述	获取 Debug 下 ET4 使能状态
输入参数 1	无
输出参数	无
返回值	Debug 下 ET4 使能: DISABLE: 禁止 ET4 ENABLE: 使能 ET4

### 8.2.10 SCU\_MCUDBGCR\_DBG\_BT1\_STOP\_Setable

函数名	SCU_MCUDBGCR_DBG_BT1_STOP_Setable
函数原型	void SCU_MCUDBGCR_DBG_BT1_STOP_Setable(FunState NewState)
功能描述	Debug 下设置 BT1 使能

输入参数 1	Debug 下 BT1 使能: DISABLE: 禁止 BT1 ENABLE: 使能 BT1
输出参数	无
返回值	无

### 8.2.11 SCU\_MCUDBGCR\_DBG\_BT1\_STOP\_Getable

函数名	SCU_MCUDBGCR_DBG_BT1_STOP_Getable
函数原型	FunState SCU_MCUDBGCR_DBG_BT1_STOP_Getable(void)
功能描述	获取 Debug 下 BT1 使能状态
输入参数 1	无
输出参数	无
返回值	Debug 下 BT1 使能: DISABLE: 禁止 BT1 ENABLE: 使能 BT1

### 8.2.12 SCU\_MCUDBGCR\_DBG\_BT2\_STOP\_Setable

函数名	SCU_MCUDBGCR_DBG_BT2_STOP_Setable
函数原型	void SCU_MCUDBGCR_DBG_BT2_STOP_Setable(FunState NewState)
功能描述	Debug 下设置 BT2 使能
输入参数 1	Debug 下 BT2 使能: DISABLE: 禁止 BT2 ENABLE: 使能 BT2
输出参数	无
返回值	无

### 8.2.13 SCU\_MCUDBGCR\_DBG\_BT2\_STOP\_Getable

函数名	SCU_MCUDBGCR_DBG_BT2_STOP_Getable
函数原型	FunState SCU_MCUDBGCR_DBG_BT2_STOP_Getable(void)
功能描述	获取 Debug 下 BT2 使能状态
输入参数 1	无
输出参数	无
返回值	Debug 下 BT2 使能: DISABLE: 禁止 BT2 ENABLE: 使能 BT2



### 8.2.14 SCU\_MCUDBGCR\_DBG\_WWDT\_STOP\_Setable

函数名	SCU_MCUDBGCR_DBG_WWDT_STOP_Setable
函数原型	void SCU_MCUDBGCR_DBG_WWDT_STOP_Setable(FunState NewState)
功能描述	Debug 下设置 WWDT 使能
输入参数 1	Debug 下 WWDT 使能: DISABLE: 禁止 WWDT ENABLE: 使能 WWDT
输出参数	无
返回值	无

### 8.2.15 SCU\_MCUDBGCR\_DBG\_WWDT\_STOP\_Getable

函数名	SCU_MCUDBGCR_DBG_WWDT_STOP_Getable
函数原型	FunState SCU_MCUDBGCR_DBG_WWDT_STOP_Getable(void)
功能描述	获取 Debug 下 WWDT 使能状态
输入参数 1	无
输出参数	无
返回值	Debug 下 WWDT 使能: DISABLE: 禁止 WWDT ENABLE: 使能 WWDT

### 8.2.16 SCU\_MCUDBGCR\_DBG\_IWDT\_STOP\_Setable

函数名	SCU_MCUDBGCR_DBG_IWDT_STOP_Setable
函数原型	void SCU_MCUDBGCR_DBG_IWDT_STOP_Setable(FunState NewState)
功能描述	Debug 下设置 IWDT 使能
输入参数 1	Debug 下 IWDT 使能: DISABLE: 禁止 IWDT ENABLE: 使能 IWDT
输出参数	无
返回值	无

### 8.2.17 SCU\_MCUDBGCR\_DBG\_IWDT\_STOP\_Getable

函数名	SCU_MCUDBGCR_DBG_IWDT_STOP_Getable
函数原型	FunState SCU_MCUDBGCR_DBG_IWDT_STOP_Getable(void)

功能描述	获取 Debug 下 IWDT 使能状态
输入参数 1	无
输出参数	无
返回值	Debug 下 IWDT 使能: DISABLE: 禁止 IWDT ENABLE: 使能 IWDT

### 8.2.18 SCU\_HDFFLAG\_HDF\_REQUEST\_FLAG\_Clr

函数名	SCU_HDFFLAG_HDF_REQUEST_FLAG_Clr
函数原型	void SCU_HDFFLAG_HDF_REQUEST_FLAG_Clr(void)
功能描述	清除 HardFault 标志
输入参数 1	无
输出参数	无
返回值	无

### 8.2.19 SCU\_HDFFLAG\_HDF\_REQUEST\_FLAG\_Chk

函数名	SCU_HDFFLAG_HDF_REQUEST_FLAG_Chk
函数原型	FlagStatus SCU_HDFFLAG_HDF_REQUEST_FLAG_Chk(void)
功能描述	获取 HardFault 标志
输入参数 1	无
输出参数	无
返回值	HardFault 标志: 0: 没产生 HardFault 1: 产生 HardFault

### 8.2.20 SCU\_HDFFLAG\_DABORT\_ADDR\_FLAG\_Clr

函数名	SCU_HDFFLAG_DABORT_ADDR_FLAG_Clr
函数原型	void SCU_HDFFLAG_DABORT_ADDR_FLAG_Clr(void)
功能描述	清除地址非对齐访问错误标志
输入参数 1	无
输出参数	无
返回值	无

### 8.2.21 SCU\_HDFFLAG\_DABORT\_ADDR\_FLAG\_Chk

函数名	SCU_HDFFLAG_DABORT_ADDR_FLAG_Chk
函数原型	FlagStatus SCU_HDFFLAG_DABORT_ADDR_FLAG_Chk(void)
功能描述	获取地址非对齐访问错误标志
输入参数 1	无
输出参数	无
返回值	地址非对齐访问错误标志： 0：没产生地址非对齐访问错误 1：产生地址非对齐访问错误

### 8.2.22 SCU\_HDFFLAG\_DABORT\_RESP\_FLAG\_Clr

函数名	SCU_HDFFLAG_DABORT_RESP_FLAG_Clr
函数原型	void SCU_HDFFLAG_DABORT_RESP_FLAG_Clr(void)
功能描述	清除非法地址访问错误标志
输入参数 1	无
输出参数	无
返回值	无

### 8.2.23 SCU\_HDFFLAG\_DABORT\_RESP\_FLAG\_Chk

函数名	SCU_HDFFLAG_DABORT_RESP_FLAG_Chk
函数原型	FlagStatus SCU_HDFFLAG_DABORT_RESP_FLAG_Chk(void)
功能描述	获取非法地址访问错误标志
输入参数 1	无
输出参数	无
返回值	非法地址访问错误标志： 0：没产生非法地址访问错误 1：产生非法地址访问错误

### 8.2.24 SCU\_HDFFLAG\_SVCUNDEF\_FLAG\_Clr

函数名	SCU_HDFFLAG_SVCUNDEF_FLAG_Clr
函数原型	void SCU_HDFFLAG_SVCUNDEF_FLAG_Clr(void)
功能描述	清除 SVC instructions 未定义标志
输入参数 1	无
输出参数	无
返回值	无

**8.2.25 SCU\_HDFFLAG\_SVCUNDEF\_FLAG\_Chk**

函数名	SCU_HDFFLAG_SVCUNDEF_FLAG_Chk
函数原型	FlagStatus SCU_HDFFLAG_SVCUNDEF_FLAG_Chk(void)
功能描述	获取 SVC instructions 未定义标志
输入参数 1	无
输出参数	无
返回值	SVC instructions 未定义标志： 0: 没产生 SVC instructions 未定义错误 1: 产生 SVC instructions 未定义错误

**8.2.26 SCU\_HDFFLAG\_BKPT\_FLAG\_Clr**

函数名	SCU_HDFFLAG_BKPT_FLAG_Clr
函数原型	void SCU_HDFFLAG_BKPT_FLAG_Clr(void)
功能描述	清除执行 BKPT 指令标志
输入参数 1	无
输出参数	无
返回值	无

**8.2.27 SCU\_HDFFLAG\_BKPT\_FLAG\_Chk**

函数名	SCU_HDFFLAG_BKPT_FLAG_Chk
函数原型	FlagStatus SCU_HDFFLAG_BKPT_FLAG_Chk(void)
功能描述	获取执行 BKPT 指令标志
输入参数 1	无
输出参数	无
返回值	执行 BKPT 指令标志

**8.2.28 SCU\_HDFFLAG\_TBIT\_FLAG\_Clr**

函数名	SCU_HDFFLAG_TBIT_FLAG_Clr
函数原型	void SCU_HDFFLAG_TBIT_FLAG_Clr(void)
功能描述	清除 Thumb-State 标志
输入参数 1	无
输出参数	无
返回值	无



### 8.2.29 SCU\_HDFFLAG\_TBIT\_FLAG\_Chk

函数名	SCU_HDFFLAG_TBIT_FLAG_Chk
函数原型	FlagStatus SCU_HDFFLAG_TBIT_FLAG_Chk(void)
功能描述	获取 Thumb-State 标志
输入参数 1	无
输出参数	无
返回值	Thumb-State 标志

### 8.2.30 SCU\_HDFFLAG\_SPECIAL\_OP\_FLAG\_Clr

函数名	SCU_HDFFLAG_SPECIAL_OP_FLAG_Clr
函数原型	void SCU_HDFFLAG_SPECIAL_OP_FLAG_Clr(void)
功能描述	清除特殊指令标志
输入参数 1	无
输出参数	无
返回值	无

### 8.2.31 SCU\_HDFFLAG\_SPECIAL\_OP\_FLAG\_Chk

函数名	SCU_HDFFLAG_SPECIAL_OP_FLAG_Chk
函数原型	FlagStatus SCU_HDFFLAG_SPECIAL_OP_FLAG_Chk(void)
功能描述	获取特殊指令标志
输入参数 1	无
输出参数	无
返回值	特殊指令标志

## 9 ANAC

### 9.1 ANAC 寄存器结构

寄存器	描述
PDRCON	PDR 设置寄存器
BORCON	BOR 设置寄存器
SVDCFG	SVD 设置寄存器
SVDCON	SVD 控制寄存器
SVDSIF	SVD 状态和标志寄存器
FDETIE	停振检测中断使能寄存器
FDETIF	停振检测中断标志寄存器
ADCINSEL	ADC 输入通道选择寄存器
ADCCON	ADC 控制寄存器
ADCTRIM	ADC 调校寄存器
ADCDATA	ADC 输出数据寄存器
ADCIF	ADC 中断标志寄存器
COMP1CR	COMP1 控制寄存器
COMP2CR	COMP2 控制寄存器
COMPICR	COMP 中断设置寄存器
COMPIF	COMP 中断标志寄存器
SVDALM	SVD 电压检测标志寄存器

### 9.2 ANAC 库函数

序号	函数名	描述
1	ANAC_Deinit	ANAC 初始化为缺省值
2	ANAC_PDRCON_PDRCFG_Set	设置 PDR 下电复位电压
3	ANAC_PDRCON_PDRCFG_Get	获取 PDR 下电复位电压
4	ANAC_PDRCON_PDREN_Setable	设置 PDR 下电复位使能
5	ANAC_PDRCON_PDREN_Getable	获取 PDR 下电复位使能状态
6	ANAC_BORCON_BOR_PDRCFG_Set	设置 BOR 下电复位电压
7	ANAC_BORCON_BOR_PDRCFG_Get	获取 BOR 下电复位电压
8	ANAC_BORCON_OFF_BOR_Setable	设置 BOR 下电复位使能
9	ANAC_BORCON_OFF_BOR_Getable	获取 BOR 下电复位使能状态

10	ANAC_SVD_Init	SVD 初始化设置
11	ANAC_SVDCON_SVDEN_Setable	设置 SVD 使能
12	ANAC_SVDCON_SVDEN_Getable	获取 SVD 使能状态
13	ANAC_SVDCFG_SVDMOD_Set	设置 SVD 工作模式
14	ANAC_SVDCFG_SVDMOD_Get	获取 SVD 工作模式
15	ANAC_SVDCFG_PFIE_Setable	设置 SVD 电源跌落中断使能
16	ANAC_SVDCFG_PFIE_Getable	获取 SVD 电源跌落中断使能状态
17	ANAC_SVDCFG_PRIIE_Setable	设置 SVD 电源恢复中断使能
18	ANAC_SVDCFG_PRIIE_Getable	获取 SVD 电源恢复中断使能状态
19	ANAC_SVDCFG_SVDLVL_Set	设置 SVD 报警阈值
20	ANAC_SVDCFG_SVDLVL_Get	获取 SVD 报警阈值
21	ANAC_SVDCFG_DFEN_Setable	设置 SVD 数字滤波使能
22	ANAC_SVDCFG_DFEN_Getable	获取 SVD 数字滤波使能状态
23	ANAC_SVDCFG_SVDITVL_Set	设置 SVD 间歇使能间隔
24	ANAC_SVDCFG_SVDITVL_Get	获取 SVD 间歇使能间隔
25	ANAC_SVDCON_SVDTE_Setable	设置 SVD 测试使能
26	ANAC_SVDCON_SVDTE_Getable	获取 SVD 测试使能状态
27	ANAC_SVDSIF_SVDO_Chk	获取 SVD 电源检测输出状态
28	ANAC_SVDSIF_PFF_Clr	清除 SVD 电源跌落中断标志
29	ANAC_SVDSIF_PFF_Chk	获取 SVD 电源跌落中断标志状态
30	ANAC_SVDSIF_PRIF_Clr	清除 SVD 电源恢复中断标志
31	ANAC_SVDSIF_PRIF_Chk	获取 SVD 电源恢复中断标志状态
32	ANAC_SVDALM_SVDALM_B_Chk	获取 SVD 内部滤波后的电压检测标志状态
33	ANAC_FDETIF_FDETO_Chk	获取停振检测模块输出状态
34	ANAC_FDETIE_FDET_IE_Setable	设置 XTLF 停振检测报警中断使能
35	ANAC_FDETIE_FDET_IE_Getable	获取 XTLF 停振检测报警中断使能状态
36	ANAC_FDETIF_FDETIF_Clr	清除停振检测中断标志
37	ANAC_FDETIF_FDETIF_Chk	获取停振检测中断标志状态
38	ANAC_ADC_Init	ADC 初始化
39	ANAC_ADCCON_ADC_EN_Setable	设置 ADC 使能
40	ANAC_ADCCON_ADC_EN_Getable	获取 ADC 使能状态
41	ANAC_ADCCON_ADC_IE_Setable	设置 ADC 中断使能
42	ANAC_ADCCON_ADC_IE_Getable	获得 ADC 中断使能状态
43	ANAC_ADCCON_ADC_VANA_EN_Set	设置 ADC 通道选择
44	ANAC_ADCCON_ADC_VANA_EN_Get	获取 ADC 通道选择





45	ANAC_ADCINSEL_BUFSEL_Set	设置 ADC 输入通道选择
46	ANAC_ADCINSEL_BUFSEL_Get	获取 ADC 输入通道选择
47	ANAC_ADCTRIM_Write	写入 ADC 调校值
48	ANAC_ADCTRIM_Read	读取 ADC 调校值
49	ANAC_ADCDATA_Read	读取 ADC 输出数据
50	ANAC_ADCIF_ADC_IF_Clr	清除 ADC 转换完成中断标志
51	ANAC_ADCIF_ADC_IF_Chk	获取 ADC 转换完成中断标志状态
52	ANAC_ADCINSEL_BUFEN_Setable	设置 ADC 输入通道 Buffer 使能
53	ANAC_ADCINSEL_BUFEN_Getable	获取 ADC 输入通道 Buffer 使能状态
54	ANAC_ADCINSEL_BUFBYP_Setable	设置 ADC 输入通道 Buffer 是否 Bypass
55	ANAC_ADCINSEL_BUFBYP_Getable	获取 ADC 输入通道 Buffer 是否 Bypass 状态
56	ANAC_ADC_Channel_SetEx	设置 ADC 测量通道
57	ANAC_ADC_VoltageCalc	ADC 电压计算
58	ANAC_ADC_TemperatureCalc	ADC 温度计算
59	ANAC_COMPx_Init	比较器初始化
60	ANAC_COMP1CR_CMP1O_Chk	读取比较器 1 输出高低
61	ANAC_COMP1CR_V1PSEL_Set	设置比较器 1 正极选择输入
62	ANAC_COMP1CR_V1PSEL_Get	获取比较器 1 正极选择输入
63	ANAC_COMP1CR_V1NSEL_Set	设置比较器 1 负极选择输入
64	ANAC_COMP1CR_V1NSEL_Get	获取比较器 1 负极选择输入
65	ANAC_COMP1CR_CMP1EN_Setable	设置比较器 1 使能
66	ANAC_COMP1CR_CMP1EN_Getable	获取比较器 1 使能状态
67	ANAC_COMP1CR_CMP1DF_Setable	设置比较器 1 数字滤波使能
68	ANAC_COMP1CR_CMP1DF_Getable	获取比较器 1 数字滤波使能状态
69	ANAC_COMP1CR_CMP1SEL_Set	设置比较器 1 中断源选择
70	ANAC_COMP1CR_CMP1SEL_Get	获取比较器 1 中断源选择
71	ANAC_COMP1CR_CMP1IE_Setable	设置比较器 1 中断使能
72	ANAC_COMP1CR_CMP1IE_Getable	获取比较器 1 中断使能状态
73	ANAC_COMP1IF_CMP1IF_Clr	清除比较器 1 中断标志
74	ANAC_COMP1IF_CMP1IF_Chk	获取比较器 1 中断标志状态
75	ANAC_COMP2CR_CMP2O_Chk	读取比较器 2 输出高低
76	ANAC_COMP2CR_V2PSEL_Set	设置比较器 2 正极选择输入
77	ANAC_COMP2CR_V2PSEL_Get	获取比较器 2 正极选择输入
78	ANAC_COMP2CR_V2NSEL_Set	设置比较器 2 负极选择输入



79	ANAC_COMP2CR_V2NSEL_Get	获取比较器 2 负极选择输入
80	ANAC_COMP2CR_CMP2EN_Setable	设置比较器 2 使能
81	ANAC_COMP2CR_CMP2EN_Getable	获取比较器 2 使能状态
82	ANAC_COMPICR_CMP2DF_Setable	设置比较器 2 数字滤波使能
83	ANAC_COMPICR_CMP2DF_Getable	获取比较器 2 数字滤波使能状态
84	ANAC_COMPICR_CMP2SEL_Set	设置比较器 2 中断源选择
85	ANAC_COMPICR_CMP2SEL_Get	获取比较器 2 中断源选择
86	ANAC_COMPICR_CMP2IE_Setable	设置比较器 2 中断使能
87	ANAC_COMPICR_CMP2IE_Getable	获取比较器 2 中断使能状态
88	ANAC_COMPIF_CMP2IF_Clr	清除比较器 2 中断标志
89	ANAC_COMPIF_CMP2IF_Chk	获取比较器 2 中断标志状态
90	ANAC_COMPICR_BUFBYP_Setable	设置比较器 Buffer Bypass 使能
91	ANAC_COMPICR_BUFBYP_Getable	获取比较器 Buffer Bypass 使能状态
92	ANAC_COMPICR_BUFENB_Setable	设置比较器 Buffer 使能
93	ANAC_COMPICR_BUFENB_Getable	获取比较器 Buffer 使能状态

### 9.2.1 ANAC\_Deinit

函数名	ANAC_Deinit
函数原型	void ANAC_Deinit(void)
功能描述	ANAC 初始化为缺省值
输入参数 1	无
输出参数	无
返回值	无

### 9.2.2 ANAC\_PDRCON\_PDRCFG\_Set

函数名	ANAC_PDRCON_PDRCFG_Set
函数原型	void ANAC_PDRCON_PDRCFG_Set(uint32_t SetValue)
功能描述	设置 PDR 下电复位电压
输入参数 1	PDR 下电复位电压: 00: 1.5V 01: 1.25V 10: 1.35V 11: 1.4V
输出参数	无
返回值	无

### 9.2.3 ANAC\_PDRCON\_PDRCFG\_Get

函数名	ANAC_PDRCON_PDRCFG_Get
函数原型	uint32_t ANAC_PDRCON_PDRCFG_Get(void)
功能描述	获取 PDR 下电复位电压
输入参数 1	无
输出参数	无
返回值	PDR 下电复位电压： 00: 1.5V 01: 1.25V 10: 1.35V 11: 1.4V

### 9.2.4 ANAC\_PDRCON\_PDREN\_Setable

函数名	ANAC_PDRCON_PDREN_Setable
函数原型	void ANAC_PDRCON_PDREN_Setable(FunState NewState)
功能描述	设置 PDR 下电复位使能
输入参数 1	PDR 下电复位使能状态： DISABLE: 禁止 PDR 下电复位 ENABLE: 使能 PDR 下电复位
输出参数	无
返回值	无

### 9.2.5 ANAC\_PDRCON\_PDREN\_Getable

函数名	ANAC_PDRCON_PDREN_Getable
函数原型	FunState ANAC_PDRCON_PDREN_Getable(void)
功能描述	获取 PDR 下电复位使能状态
输入参数 1	无
输出参数	无
返回值	PDR 下电复位使能状态： DISABLE: 禁止 PDR 下电复位 ENABLE: 使能 PDR 下电复位

### 9.2.6 ANAC\_BORCON\_BOR\_PDRCFG\_Set

函数名	ANAC_BORCON_BOR_PDRCFG_Set
函数原型	void ANAC_BORCON_BOR_PDRCFG_Set(uint32_t SetValue)
功能描述	设置 BOR 下电复位电压
输入参数 1	BOR 下电复位电压： 00: 1.5V 01: 1.25V 10: 1.35V 11: 1.4V
输出参数	无
返回值	无

### 9.2.7 ANAC\_BORCON\_BOR\_PDRCFG\_Get

函数名	ANAC_BORCON_BOR_PDRCFG_Get
函数原型	uint32_t ANAC_BORCON_BOR_PDRCFG_Get(void)
功能描述	获取 BOR 下电复位电压
输入参数 1	无
输出参数	无
返回值	BOR 下电复位电压： 00: 1.5V 01: 1.25V 10: 1.35V 11: 1.4V

### 9.2.8 ANAC\_BORCON\_OFF\_BOR\_Setable

函数名	ANAC_BORCON_OFF_BOR_Setable
函数原型	
功能描述	设置 BOR 下电复位使能
输入参数 1	BOR 下电复位使能状态： DISABLE: 禁止 BOR 下电复位 ENABLE: 使能 BOR 下电复位
输出参数	无
返回值	无

### 9.2.9 ANAC\_BORCON\_OFF\_BOR\_Getable

函数名	ANAC_BORCON_OFF_BOR_Getable
函数原型	
功能描述	获取 BOR 下电复位使能状态
输入参数 1	无
输出参数	无
返回值	BOR 下电复位使能状态： DISABLE：禁止 BOR 下电复位 ENABLE：使能 BOR 下电复位

### 9.2.10 ANAC\_SVD\_Init

函数名	ANAC_SVD_Init		
函数原型	void ANAC_SVD_Init(ANAC_SVD_InitTypeDef* para)		
功能描述	SVD 初始化设置		
输入参数 1	SVD 设置参数：		
	SVD 工作模式： 0：常使能模式 1：间歇使能模式		
	SVD 间歇使能间隔： 00：15.625ms 01：62.5ms 10：256ms 11：1s		
	SVD 报警阈值设置：		
	SVDLVL	下降阈值	上升阈值
	0000	1.800V	1.900V
	0001	2.014V	2.114V
	0010	2.229V	2.329V
	0011	2.443V	2.543V
	0100	2.657V	2.757V
	0101	2.871V	2.971V
	0110	3.086V	3.186V
	0111	3.300V	3.400V
	1000	3.514V	3.614V
	1001	3.729V	3.829V
	1010	3.943V	4.043V
1011	4.157V	4.257V	
1100	4.371V	4.471V	
1101	4.586V	4.686V	

	1110	4.800V	4.900V
	1111	SVS	SVS
	SVD 数字滤波使能:		
	DISABLE: 禁止 SVD 输出的数字滤波		
	ENABLE: 使能 SVD 输出的数字滤波		
	SVD 电源跌落中断使能:		
输出参数	DISABLE: 禁止 SVD 电源跌落中断		
	ENABLE: 使能 SVD 电源跌落中断		
	SVD 电源恢复中断使能:		
	DISABLE: 禁止 SVD 电源恢复中断		
	ENABLE: 使能 SVD 电源恢复中断		
	SVD 使能:		
返回值	DISABLE: 禁止 SVD		
	ENABLE: 使能 SVD		

### 9.2.11 ANAC\_SVDCON\_SVDEN\_Setable

函数名	ANAC_SVDCON_SVDEN_Setable
函数原型	void ANAC_SVDCON_SVDEN_Setable(FunState NewState)
功能描述	设置 SVD 使能
输入参数 1	SVD 使能: DISABLE: 禁止 SVD ENABLE: 使能 SVD
输出参数	无
返回值	无

### 9.2.12 ANAC\_SVDCON\_SVDEN\_Getable

函数名	ANAC_SVDCON_SVDEN_Getable
函数原型	FunState ANAC_SVDCON_SVDEN_Getable(void)
功能描述	获取 SVD 使能状态
输入参数 1	无
输出参数	无
返回值	SVD 使能: DISABLE: 禁止 SVD ENABLE: 使能 SVD

### 9.2.13 ANAC\_SVDCFG\_SVDMOD\_Set

函数名	ANAC_SVDCFG_SVDMOD_Set
函数原型	void ANAC_SVDCFG_SVDMOD_Set(uint32_t SetValue)
功能描述	设置 SVD 工作模式
输入参数 1	SVD 工作模式： 0：常使能模式 1：间歇使能模式
输出参数	无
返回值	无

### 9.2.14 ANAC\_SVDCFG\_SVDMOD\_Get

函数名	ANAC_SVDCFG_SVDMOD_Get
函数原型	uint32_t ANAC_SVDCFG_SVDMOD_Get(void)
功能描述	获取 SVD 工作模式
输入参数 1	无
输出参数	无
返回值	SVD 工作模式： 0：常使能模式 1：间歇使能模式

### 9.2.15 ANAC\_SVDCFG\_PFIE\_Setable

函数名	ANAC_SVDCFG_PFIE_Setable
函数原型	void ANAC_SVDCFG_PFIE_Setable(FunState NewState)
功能描述	设置 SVD 电源跌落中断使能
输入参数 1	SVD 电源跌落中断使能： DISABLE：禁止 SVD 电源跌落中断 ENABLE：使能 SVD 电源跌落中断
输出参数	无
返回值	无

### 9.2.16 ANAC\_SVDCFG\_PFIE\_Getable

函数名	ANAC_SVDCFG_PFIE_Getable
函数原型	FunState ANAC_SVDCFG_PFIE_Getable(void)
功能描述	获取 SVD 电源跌落中断使能状态

输入参数 1	无
输出参数	无
返回值	SVD 电源跌落中断使能： DISABLE：禁止 SVD 电源跌落中断 ENABLE：使能 SVD 电源跌落中断

### 9.2.17 ANAC\_SVDCFG\_PRIE\_Setable

函数名	ANAC_SVDCFG_PRIE_Setable
函数原型	void ANAC_SVDCFG_PRIE_Setable(FunState NewState)
功能描述	设置 SVD 电源恢复中断使能
输入参数 1	SVD 电源恢复中断使能： DISABLE：禁止 SVD 电源恢复中断 ENABLE：使能 SVD 电源恢复中断
输出参数	无
返回值	无

### 9.2.18 ANAC\_SVDCFG\_PRIE\_Getable

函数名	ANAC_SVDCFG_PRIE_Getable
函数原型	FunState ANAC_SVDCFG_PRIE_Getable(void)
功能描述	获取 SVD 电源恢复中断使能状态
输入参数 1	无
输出参数	无
返回值	SVD 电源恢复中断使能： DISABLE：禁止 SVD 电源恢复中断 ENABLE：使能 SVD 电源恢复中断

### 9.2.19 ANAC\_SVDCFG\_SVDLVL\_Set

函数名	ANAC_SVDCFG_SVDLVL_Set		
函数原型	void ANAC_SVDCFG_SVDLVL_Set(uint32_t SetValue)		
功能描述	设置 SVD 报警阈值		
输入参数 1	SVD 报警阈值设置：		
	SVDLVL	下降阈值	上升阈值
	0000	1.800V	1.900V
	0001	2.014V	2.114V
	0010	2.229V	2.329V



	0011	2.443V	2.543V
	0100	2.657V	2.757V
	0101	2.871V	2.971V
	0110	3.086V	3.186V
	0111	3.300V	3.400V
	1000	3.514V	3.614V
	1001	3.729V	3.829V
	1010	3.943V	4.043V
	1011	4.157V	4.257V
	1100	4.371V	4.471V
	1101	4.586V	4.686V
	1110	4.800V	4.900V
	1111	SVS	SVS
输出参数	无		
返回值	无		

### 9.2.20 ANAC\_SVDCFG\_SVDLVL\_Get

函数名	ANAC_SVDCFG_SVDLVL_Get		
函数原型	uint32_t ANAC_SVDCFG_SVDLVL_Get(void)		
功能描述	获取 SVD 报警阈值		
输入参数 1	无		
输出参数	无		
返回值	SVD 报警阈值设置:		
	SVDLVL	下降阈值	上升阈值
	0000	1.800V	1.900V
	0001	2.014V	2.114V
	0010	2.229V	2.329V
	0011	2.443V	2.543V
	0100	2.657V	2.757V
	0101	2.871V	2.971V
	0110	3.086V	3.186V
	0111	3.300V	3.400V
	1000	3.514V	3.614V
	1001	3.729V	3.829V
	1010	3.943V	4.043V
	1011	4.157V	4.257V
	1100	4.371V	4.471V
	1101	4.586V	4.686V
	1110	4.800V	4.900V
	1111	SVS	SVS



### 9.2.21 ANAC\_SVDCFG\_DFEN\_Setable

函数名	ANAC_SVDCFG_DFEN_Setable
函数原型	void ANAC_SVDCFG_DFEN_Setable(FunState NewState)
功能描述	设置 SVD 数字滤波使能
输入参数 1	SVD 数字滤波使能： DISABLE：禁止 SVD 输出的数字滤波 ENABLE：使能 SVD 输出的数字滤波
输出参数	无
返回值	无

### 9.2.22 ANAC\_SVDCFG\_DFEN\_Getable

函数名	ANAC_SVDCFG_DFEN_Getable
函数原型	FunState ANAC_SVDCFG_DFEN_Getable(void)
功能描述	获取 SVD 数字滤波使能状态
输入参数 1	无
输出参数	无
返回值	SVD 数字滤波使能： DISABLE：禁止 SVD 输出的数字滤波 ENABLE：使能 SVD 输出的数字滤波

### 9.2.23 ANAC\_SVDCFG\_SVDITVL\_Set

函数名	ANAC_SVDCFG_SVDITVL_Set
函数原型	void ANAC_SVDCFG_SVDITVL_Set(uint32_t SetValue)
功能描述	设置 SVD 间歇使能间隔
输入参数 1	SVD 间歇使能间隔： 00: 15.625ms 01: 62.5ms 10: 256ms 11: 1s
输出参数	无
返回值	无

### 9.2.24 ANAC\_SVDCFG\_SVDITVL\_Get

函数名	ANAC_SVDCFG_SVDITVL_Get
函数原型	uint32_t ANAC_SVDCFG_SVDITVL_Get(void)
功能描述	获取 SVD 间歇使能间隔
输入参数 1	无
输出参数	无
返回值	SVD 间歇使能间隔： 00: 15.625ms 01: 62.5ms 10: 256ms 11: 1s

### 9.2.25 ANAC\_SVDCON\_SVDTE\_Setable

函数名	ANAC_SVDCON_SVDTE_Setable
函数原型	void ANAC_SVDCON_SVDTE_Setable(FunState NewState)
功能描述	设置 SVD 测试使能
输入参数 1	SVD 测试使能： DISABLE: 禁止 SVD 测试 ENABLE: 使能 SVD 测试
输出参数	无
返回值	无

### 9.2.26 ANAC\_SVDCON\_SVDTE\_Getable

函数名	ANAC_SVDCON_SVDTE_Getable
函数原型	FunState ANAC_SVDCON_SVDTE_Getable(void)
功能描述	获取 SVD 测试使能状态
输入参数 1	无
输出参数	无
返回值	SVD 测试使能： DISABLE: 禁止 SVD 测试 ENABLE: 使能 SVD 测试

### 9.2.27 ANAC\_SVDSIF\_SVDO\_Chk

函数名	ANAC_SVDSIF_SVDO_Chk
-----	----------------------

函数原型	FlagStatus ANAC_SVDSIF_SVDO_Chk(void)
功能描述	获取 SVD 电源检测输出状态
输入参数 1	无
输出参数	无
返回值	SVD 电源检测输出状态： 0：电源电压低于 SVD 当前阈值 1：电源电压高于 SVD 当前阈值

### 9.2.28 ANAC\_SVDSIF\_PFF\_Clr

函数名	ANAC_SVDSIF_PFF_Clr
函数原型	void ANAC_SVDSIF_PFF_Clr(void)
功能描述	清除 SVD 电源跌落中断标志
输入参数 1	无
输出参数	无
返回值	无

### 9.2.29 ANAC\_SVDSIF\_PFF\_Chk

函数名	ANAC_SVDSIF_PFF_Chk
函数原型	FlagStatus ANAC_SVDSIF_PFF_Chk(void)
功能描述	获取 SVD 电源跌落中断标志状态
输入参数 1	无
输出参数	无
返回值	SVD 电源跌落中断标志： 0：未发生 SVD 电源跌落中断 1：发生 SVD 电源跌落中断

### 9.2.30 ANAC\_SVDSIF\_PRF\_Clr

函数名	ANAC_SVDSIF_PRF_Clr
函数原型	void ANAC_SVDSIF_PRF_Clr(void)
功能描述	清除 SVD 电源恢复中断标志
输入参数 1	无
输出参数	无
返回值	无

### 9.2.31 ANAC\_SVDSIF\_PRF\_Chk

函数名	ANAC_SVDSIF_PRF_Chk
函数原型	FlagStatus ANAC_SVDSIF_PRF_Chk(void)
功能描述	获取 SVD 电源恢复中断标志状态
输入参数 1	无
输出参数	无
返回值	SVD 电源恢复中断标志： 0: 未发生 SVD 电源恢复中断 1: 发生 SVD 电源恢复中断

### 9.2.32 ANAC\_SVDALM\_SVDALM\_B\_Chk

函数名	ANAC_SVDALM_SVDALM_B_Chk
函数原型	FlagStatus ANAC_SVDALM_SVDALM_B_Chk(void)
功能描述	获取 SVD 内部滤波后的电压检测标志状态
输入参数 1	无
输出参数	无
返回值	SVD 内部滤波后的电压检测标志状态： 0: SVD 内部滤波后不欠压 1: SVD 内部滤波后欠压

### 9.2.33 ANAC\_FDETIF\_FDETO\_Chk

函数名	ANAC_FDETIF_FDETO_Chk
函数原型	FlagStatus ANAC_FDETIF_FDETO_Chk(void)
功能描述	获取停振检测模块输出状态
输入参数 1	无
输出参数	无
返回值	停振检测模块输出状态： 0: XTIF 未停振 1: XTIF 停振

### 9.2.34 ANAC\_FDETIE\_FDET\_IE\_Setable

函数名	ANAC_FDETIE_FDET_IE_Setable
函数原型	void ANAC_FDETIE_FDET_IE_Setable(FunState NewState)
功能描述	设置 XTIF 停振检测报警中断使能

输入参数 1	XTLF 停振检测报警中断使能状态： DISABLE：禁止 XTLF 停振检测报警中断 ENABLE：使能 XTLF 停振检测报警中断
输出参数	无
返回值	无

### 9.2.35 ANAC\_FDETIE\_FDET\_IE\_Getable

函数名	ANAC_FDETIE_FDET_IE_Getable
函数原型	FunState ANAC_FDETIE_FDET_IE_Getable(void)
功能描述	获取 XTLF 停振检测报警中断使能状态
输入参数 1	无
输出参数	无
返回值	XTLF 停振检测报警中断使能状态： DISABLE：禁止 XTLF 停振检测报警中断 ENABLE：使能 XTLF 停振检测报警中断

### 9.2.36 ANAC\_FDETIF\_FDETIF\_Clr

函数名	ANAC_FDETIF_FDETIF_Clr
函数原型	void ANAC_FDETIF_FDETIF_Clr(void)
功能描述	清除停振检测中断标志
输入参数 1	无
输出参数	无
返回值	无

### 9.2.37 ANAC\_FDETIF\_FDETIF\_Chk

函数名	ANAC_FDETIF_FDETIF_Chk
函数原型	FlagStatus ANAC_FDETIF_FDETIF_Chk(void)
功能描述	获取停振检测中断标志状态
输入参数 1	无
输出参数	无
返回值	XTLF 停振检测中断标志状态： 0：未产生 XTLF 停振中断 1：产生 XTLF 停振中断

### 9.2.38 ANAC\_ADC\_Init

函数名	ANAC_ADC_Init
函数原型	void ANAC_ADC_Init(ANAC_ADC_InitTypeDef* para)
功能描述	ADC 初始化
输入参数 1	<p>ADC 参数:</p> <p>ADC 调校值: 0X3FF: ADC 转电压 0X640: ADC 转温度</p> <p>ADC 内部、外部通道选择: 0: ADC 作用于温度传感器 1: ADC 作用于测量外部电压</p> <p>ADC 通道选择: 0000-0101: 保留, 禁止使用 0110: VDD 主电源 0111: 保留 1000: ADC_IN1 1001: ADC_IN2 1010: ADC_IN3 1011: ADC_IN4 1100: ADC_IN5 1101: ADC_IN6 1110: ADC_IN7 1111: ADC_IN8</p> <p>ADC 输入通道 Buffer 使能: DISABLE: 禁止 ADC 输入通道 Buffer ENABLE: 使能 ADC 输入通道 Buffer</p> <p>ADC Buffer Bypass: 0: 停止 Bypass, ADC 测量外部信号 1: 发生 Bypass, ADC 测量电源电压</p> <p>ADC 中断使能: DISABLE: 禁止 ADC 中断 ENABLE: 使能 ADC 中断</p> <p>ADC 使能: DISABLE: 禁止 ADC ENABLE: 使能 ADC</p>
输出参数	无
返回值	无

### 9.2.39 ANAC\_ADCCON\_ADC\_EN\_Setable

函数名	ANAC_ADCCON_ADC_EN_Setable
-----	----------------------------

函数原型	void ANAC_ADCCON_ADC_EN_Setable(FunState NewState)
功能描述	设置 ADC 使能
输入参数 1	ADC 使能: DISABLE: 禁止 ADC ENABLE: 使能 ADC
输出参数	无
返回值	无

#### 9.2.40 ANAC\_ADCCON\_ADC\_EN\_Getable

函数名	ANAC_ADCCON_ADC_EN_Getable
函数原型	void ANAC_ADCTRIM_Write(uint32_t SetValue)
功能描述	获取 ADC 使能状态
输入参数 1	无
输出参数	无
返回值	ADC 使能: DISABLE: 禁止 ADC ENABLE: 使能 ADC

#### 9.2.41 ANAC\_ADCCON\_ADC\_IE\_Setable

函数名	ANAC_ADCCON_ADC_IE_Setable
函数原型	void ANAC_ADCCON_ADC_IE_Setable(FunState NewState)
功能描述	设置 ADC 中断使能
输入参数 1	ADC 中断使能: DISABLE: 禁止 ADC 中断 ENABLE: 使能 ADC 中断
输出参数	无
返回值	无

#### 9.2.42 ANAC\_ADCCON\_ADC\_IE\_Getable

函数名	ANAC_ADCCON_ADC_IE_Getable
函数原型	FunState ANAC_ADCCON_ADC_IE_Getable(void)
功能描述	获得 ADC 中断使能状态
输入参数 1	无
输出参数	无
返回值	ADC 中断使能:



DISABLE: 禁止 ADC 中断

ENABLE: 使能 ADC 中断

**9.2.43 ANAC\_ADCCON\_ADC\_VANA\_EN\_Set**

函数名	ANAC_ADCCON_ADC_VANA_EN_Set
函数原型	void ANAC_ADCCON_ADC_VANA_EN_Set(uint32_t SetValue)
功能描述	设置 ADC 通道选择
输入参数 1	ADC 内部、外部通道选择: 0: ADC 作用于温度传感器 1: ADC 作用于测量外部电压
输出参数	无
返回值	无

**9.2.44 ANAC\_ADCCON\_ADC\_VANA\_EN\_Get**

函数名	ANAC_ADCCON_ADC_VANA_EN_Get
函数原型	uint32_t ANAC_ADCCON_ADC_VANA_EN_Get(void)
功能描述	获取 ADC 通道选择
输入参数 1	无
输出参数	无
返回值	ADC 内部、外部通道选择: 0: ADC 作用于温度传感器 1: ADC 作用于测量外部电压

**9.2.45 ANAC\_ADCINSEL\_BUFSEL\_Set**

函数名	ANAC_ADCINSEL_BUFSEL_Set
函数原型	void ANAC_ADCINSEL_BUFSEL_Set(uint32_t SetValue)
功能描述	设置 ADC 输入通道选择
输入参数 1	ADC 通道选择: 0000-0101: 保留, 禁止使用 0110: VDD 主电源 0111: 保留 1000: ADC_IN1 1001: ADC_IN2 1010: ADC_IN3 1011: ADC_IN4



	1100: ADC_IN5 1101: ADC_IN6 1110: ADC_IN7 1111: ADC_IN8
输出参数	无
返回值	无

### 9.2.46 ANAC\_ADCINSEL\_BUFSEL\_Get

函数名	ANAC_ADCINSEL_BUFSEL_Get
函数原型	uint32_t ANAC_ADCINSEL_BUFSEL_Get(void)
功能描述	获取 ADC 输入通道选择
输入参数 1	无
输出参数	无
返回值	ADC 通道选择: 0000-0101: 保留, 禁止使用 0110: VDD 主电源 0111: 保留 1000: ADC_IN1 1001: ADC_IN2 1010: ADC_IN3 1011: ADC_IN4 1100: ADC_IN5 1101: ADC_IN6 1110: ADC_IN7 1111: ADC_IN8

### 9.2.47 ANAC\_ADCTRIM\_Write

函数名	ANAC_ADCTRIM_Write
函数原型	void ANAC_ADCTRIM_Write(uint32_t SetValue)
功能描述	写入 ADC 调校值
输入参数 1	ADC 调校值: 0X3FF: ADC 转电压 0X640: ADC 转温度
输出参数	无
返回值	无

**9.2.48 ANAC\_ADCTRIM\_Read**

函数名	ANAC_ADCTRIM_Read
函数原型	uint32_t ANAC_ADCTRIM_Read(void)
功能描述	读取 ADC 调校值
输入参数 1	无
输出参数	无
返回值	ADC 调校值： 0X3FF: ADC 转电压 0X640: ADC 转温度

**9.2.49 ANAC\_ADCDATA\_Read**

函数名	ANAC_ADCDATA_Read
函数原型	uint32_t ANAC_ADCDATA_Read(void)
功能描述	读取 ADC 输出数据
输入参数 1	无
输出参数	无
返回值	ADC 输出数据

**9.2.50 ANAC\_ADCIF\_ADC\_IF\_Clr**

函数名	ANAC_ADCIF_ADC_IF_Clr
函数原型	void ANAC_ADCIF_ADC_IF_Clr(void)
功能描述	清除 ADC 转换完成中断标志
输入参数 1	无
输出参数	无
返回值	无

**9.2.51 ANAC\_ADCIF\_ADC\_IF\_Chk**

函数名	ANAC_ADCIF_ADC_IF_Chk
函数原型	FlagStatus ANAC_ADCIF_ADC_IF_Chk(void)
功能描述	获取 ADC 转换完成中断标志状态
输入参数 1	无
输出参数	无
返回值	ADC 转换完成中断标志状态： 0: 未发生 ADC 转换完成中断



1: 发生 ADC 转换完成中断

### 9.2.52 ANAC\_ADCINSEL\_BUFEN\_Setable

函数名	ANAC_ADCINSEL_BUFEN_Setable
函数原型	void ANAC_ADCINSEL_BUFEN_Setable(FunState NewState)
功能描述	设置 ADC 输入通道 Buffer 使能
输入参数 1	ADC 输入通道 Buffer 使能状态: DISABLE: 禁止 ADC 输入通道 Buffer ENABLE: 使能 ADC 输入通道 Buffer
输出参数	无
返回值	无

### 9.2.53 ANAC\_ADCINSEL\_BUFEN\_Getable

函数名	ANAC_ADCINSEL_BUFEN_Getable
函数原型	FunState ANAC_ADCINSEL_BUFEN_Getable(void)
功能描述	获取 ADC 输入通道 Buffer 使能状态
输入参数 1	无
输出参数	无
返回值	ADC 输入通道 Buffer 使能状态: DISABLE: 禁止 ADC 输入通道 Buffer ENABLE: 使能 ADC 输入通道 Buffer

### 9.2.54 ANAC\_ADCINSEL\_BUFBYP\_Setable

函数名	ANAC_ADCINSEL_BUFBYP_Setable
函数原型	void ANAC_ADCINSEL_BUFBYP_Setable(FunState NewState)
功能描述	设置 ADC 输入通道 Buffer 是否 Bypass
输入参数 1	ADC 输入通道 Buffer 是否 Bypass: 0: 停止 Bypass , ADC 测量外部信号 1: 发生 Bypass , ADC 测量电源电压
输出参数	无
返回值	无

### 9.2.55 ANAC\_ADCINSEL\_BUFBYP\_Getable

函数名	ANAC_ADCINSEL_BUFBYP_Getable
-----	------------------------------

函数原型	FunState ANAC_ADCINSEL_BUFBYP_Getable(void)
功能描述	获取 ADC 输入通道 Buffer 是否 Bypass 状态
输入参数 1	无
输出参数	无
返回值	ADC 输入通道 Buffer 是否 Bypass: 0: 停止 Bypass , ADC 测量外部信号 1: 发生 Bypass , ADC 测量电源电压

### 9.2.56 ANAC\_ADC\_Channel\_SetEx

函数名	ANAC_ADC_Channel_SetEx
函数原型	void ANAC_ADC_Channel_SetEx(uint8_t ChSel_def)
功能描述	设置 ADC 测量通道
输入参数 1	ADC 测量通道: 0: 温度传感器 1: 电源电压 2: CH_IN1 3: CH_IN2 4: CH_IN3 5: CH_IN4 6: CH_IN5 7: CH_IN6 8: CH_IN7 9: CH_IN8
输出参数	无
返回值	无

### 9.2.57 ANAC\_ADC\_VoltageCalc

函数名	ANAC_ADC_VoltageCalc
函数原型	float ANAC_ADC_VoltageCalc(float fADCDData)
功能描述	ADC 电压计算
输入参数 1	AD 值
输出参数	无
返回值	电压

### 9.2.58 ANAC\_ADC\_TemperatureCalc

函数名	ANAC_ADC_TemperatureCalc
函数原型	float ANAC_ADC_TemperatureCalc(float fADCData ,uint8_t Vdd)
功能描述	ADC 温度计算
输入参数 1	AD 值
输出参数	无
返回值	温度

### 9.2.59 ANAC\_COMPx\_Init

函数名	ANAC_COMPx_Init
函数原型	void ANAC_COMPx_Init(ANAC_COMPx_InitTypeDef* para)
功能描述	比较器初始化
输入参数 1	<div>比较器选择:</div> <div>1: 比较器 1</div> <div>2: 比较器 2</div> <div>比较器 1 正极输入选择:</div> <div>00: PF6</div> <div>01: PF2</div> <div>10: PG2</div> <div>11: PG3</div> <div>比较器 2 正极输入选择:</div> <div>0: PC15</div> <div>0: PE4</div> <div>比较器 1 负极输入选择:</div> <div>00: PF5</div> <div>01: PF1</div> <div>10: Vref 0.8V</div> <div>11: Vref/2 0.4V</div> <div>比较器 2 负极输入选择:</div> <div>00: PC14</div> <div>01: PE3</div> <div>10: Vref 0.8V</div> <div>11: Vref/2 0.4V</div> <div>比较器中断源选择:</div> <div>00/11: 比较器输出上升或下降沿产生中断</div> <div>01: 比较器输出上升沿产生中断</div> <div>10: 比较器输出下降沿产生中断</div> <div>比较器中断使能:</div> <div>DISABLE: 禁止比较器中断</div> <div>ENABLE: 使能比较器中断</div>

	比较器数字滤波使能： DISABLE：禁止比较器数字滤波 ENABLE：使能比较器数字滤波 比较器使能： DISABLE：禁止比较器 ENABLE：使能比较器	
输出参数	无	
返回值	无	

### 9.2.60 ANAC\_COMP1CR\_CMP10\_Chk

函数名	ANAC_COMP1CR_CMP10_Chk
函数原型	FlagStatus ANAC_COMP1CR_CMP10_Chk(void)
功能描述	读取比较器 1 输出高低
输入参数 1	无
输出参数	无
返回值	比较器 1 输出高低： 0：输出低 1：输出高

### 9.2.61 ANAC\_COMP1CR\_V1PSEL\_Set

函数名	ANAC_COMP1CR_V1PSEL_Set
函数原型	void ANAC_COMP1CR_V1PSEL_Set(uint32_t SetValue)
功能描述	设置比较器 1 正极选择输入
输入参数 1	比较器 1 正极输入选择： 00：PF6 01：PF2 10：PG2 11：PG3
输出参数	无
返回值	无

### 9.2.62 ANAC\_COMP1CR\_V1PSEL\_Get

函数名	ANAC_COMP1CR_V1PSEL_Get
函数原型	uint32_t ANAC_COMP1CR_V1PSEL_Get(void)
功能描述	获取比较器 1 正极选择输入

输入参数 1	无
输出参数	无
返回值	比较器 1 正极输入选择: 00: PF6 01: PF2 10: PG2 11: PG3

### 9.2.63 ANAC\_COMP1CR\_V1NSEL\_Set

函数名	ANAC_COMP1CR_V1NSEL_Set
函数原型	void ANAC_COMP1CR_V1NSEL_Set(uint32_t SetValue)
功能描述	设置比较器 1 负极选择输入
输入参数 1	比较器 1 负极输入选择: 00: PF5 01: PF1 10: Vref 0.8V 11: Vref/2 0.4V
输出参数	无
返回值	无

### 9.2.64 ANAC\_COMP1CR\_V1NSEL\_Get

函数名	ANAC_COMP1CR_V1NSEL_Get
函数原型	uint32_t ANAC_COMP1CR_V1NSEL_Get(void)
功能描述	获取比较器 1 负极选择输入
输入参数 1	无
输出参数	无
返回值	比较器 1 负极输入选择: 00: PF5 01: PF1 10: Vref 0.8V 11: Vref/2 0.4V

### 9.2.65 ANAC\_COMP1CR\_CMP1EN\_Setable

函数名	ANAC_COMP1CR_CMP1EN_Setable
函数原型	void ANAC_COMP1CR_CMP1EN_Setable(FunState NewState)

功能描述	设置比较器 1 使能
输入参数 1	比较器 1 使能状态： DISABLE：禁止比较器 1 ENABLE：使能比较器 1
输出参数	无
返回值	无

### 9.2.66 ANAC\_COMP1CR\_CMP1EN\_Getable

函数名	ANAC_COMP1CR_CMP1EN_Getable
函数原型	FunState ANAC_COMP1CR_CMP1EN_Getable(void)
功能描述	获取比较器 1 使能状态
输入参数 1	无
输出参数	无
返回值	比较器 1 使能状态： DISABLE：禁止比较器 1 ENABLE：使能比较器 1

### 9.2.67 ANAC\_COMPICR\_CMP1DF\_Setable

函数名	ANAC_COMPICR_CMP1DF_Setable
函数原型	void ANAC_COMPICR_CMP1DF_Setable(FunState NewState)
功能描述	设置比较器 1 数字滤波使能
输入参数 1	比较器 1 数字滤波使能： DISABLE：禁止比较器 1 数字滤波 ENABLE：使能比较器 1 数字滤波
输出参数	无
返回值	无

### 9.2.68 ANAC\_COMPICR\_CMP1DF\_Getable

函数名	ANAC_COMPICR_CMP1DF_Getable
函数原型	FunState ANAC_COMPICR_CMP1DF_Getable(void)
功能描述	获取比较器 1 数字滤波使能状态
输入参数 1	无
输出参数	无
返回值	比较器 1 数字滤波使能： DISABLE：禁止比较器 1 数字滤波



ENABLE: 使能比较器 1 数字滤波

**9.2.69 ANAC\_COMPICR\_CMP1SEL\_Set**

函数名	ANAC_COMPICR_CMP1SEL_Set
函数原型	void ANAC_COMPICR_CMP1SEL_Set(uint32_t SetValue)
功能描述	设置比较器 1 中断源选择
输入参数 1	比较器 1 中断源选择: 00/11: 比较器 1 输出上升或下降沿产生中断 01: 比较器 1 输出上升沿产生中断 10: 比较器 1 输出下降沿产生中断
输出参数	无
返回值	无

**9.2.70 ANAC\_COMPICR\_CMP1SEL\_Get**

函数名	ANAC_COMPICR_CMP1SEL_Get
函数原型	uint32_t ANAC_COMPICR_CMP1SEL_Get(void)
功能描述	获取比较器 1 中断源选择
输入参数 1	无
输出参数	无
返回值	比较器 1 中断源选择: 00/11: 比较器 1 输出上升或下降沿产生中断 01: 比较器 1 输出上升沿产生中断 10: 比较器 1 输出下降沿产生中断

**9.2.71 ANAC\_COMPICR\_CMP1IE\_Setable**

函数名	ANAC_COMPICR_CMP1IE_Setable
函数原型	void ANAC_COMPICR_CMP1IE_Setable(FunState NewState)
功能描述	设置比较器 1 中断使能
输入参数 1	比较器 1 中断使能: DISABLE: 禁止比较器 1 中断 ENABLE: 使能比较器 1 中断
输出参数	无
返回值	无

### 9.2.72 ANAC\_COMPICR\_CMP1IE\_Getable

函数名	ANAC_COMPICR_CMP1IE_Getable
函数原型	FunState ANAC_COMPICR_CMP1IE_Getable(void)
功能描述	获取比较器 1 中断使能状态
输入参数 1	无
输出参数	无
返回值	比较器 1 中断使能： DISABLE：禁止比较器 1 中断 ENABLE：使能比较器 1 中断

### 9.2.73 ANAC\_COMPIF\_CMP1IF\_Clr

函数名	ANAC_COMPIF_CMP1IF_Clr
函数原型	void ANAC_COMPIF_CMP1IF_Clr(void)
功能描述	清除比较器 1 中断标志
输入参数 1	无
输出参数	无
返回值	无

### 9.2.74 ANAC\_COMPIF\_CMP1IF\_Chk

函数名	ANAC_COMPIF_CMP1IF_Chk
函数原型	FlagStatus ANAC_COMPIF_CMP1IF_Chk(void)
功能描述	获取比较器 1 中断标志状态
输入参数 1	无
输出参数	无
返回值	比较器 1 中断标志状态： 0：未产生比较器 1 中断 1：产生比较器 1 中断标志

### 9.2.75 ANAC\_COMP2CR\_CMP2O\_Chk

函数名	ANAC_COMP2CR_CMP2O_Chk
函数原型	FlagStatus ANAC_COMP2CR_CMP2O_Chk(void)
功能描述	读取比较器 2 输出高低
输入参数 1	无
输出参数	无

返回值	比较器 2 输出高低： 0: 输出低 1: 输出高
-----	---------------------------------

### 9.2.76 ANAC\_COMP2CR\_V2PSEL\_Set

函数名	ANAC_COMP2CR_V2PSEL_Set
函数原型	void ANAC_COMP2CR_V2PSEL_Set(uint32_t SetValue)
功能描述	设置比较器 2 正极选择输入
输入参数 1	比较器 2 正极输入选择： 0: PC15 1: PE4
输出参数	无
返回值	无

### 9.2.77 ANAC\_COMP2CR\_V2PSEL\_Get

函数名	ANAC_COMP2CR_V2PSEL_Get
函数原型	uint32_t ANAC_COMP2CR_V2PSEL_Get(void)
功能描述	获取比较器 2 正极选择输入
输入参数 1	无
输出参数	无
返回值	比较器 2 正极输入选择： 0: PC15 1: PE4

### 9.2.78 ANAC\_COMP2CR\_V2NSEL\_Set

函数名	ANAC_COMP2CR_V2NSEL_Set
函数原型	void ANAC_COMP2CR_V2NSEL_Set(uint32_t SetValue)
功能描述	设置比较器 2 负极选择输入
输入参数 1	比较器 2 负极输入选择： 00: PC14 01: PE3 10: Vref 0.8V 11: Vref/2 0.4V
输出参数	无
返回值	无

### 9.2.79 ANAC\_COMP2CR\_V2NSEL\_Get

函数名	ANAC_COMP2CR_V2NSEL_Get
函数原型	uint32_t ANAC_COMP2CR_V2NSEL_Get(void)
功能描述	获取比较器 2 负极选择输入
输入参数 1	无
输出参数	无
返回值	比较器 2 负极输入选择: 00: PC14 01: PE3 10: Vref 0.8V 11: Vref/2 0.4V

### 9.2.80 ANAC\_COMP2CR\_CMP2EN\_Setable

函数名	ANAC_COMP2CR_CMP2EN_Setable
函数原型	void ANAC_COMP2CR_CMP2EN_Setable(FunState NewState)
功能描述	设置比较器 2 使能
输入参数 1	比较器 2 使能状态: DISABLE: 禁止比较器 2 ENABLE: 使能比较器 2
输出参数	无
返回值	无

### 9.2.81 ANAC\_COMP2CR\_CMP2EN\_Getable

函数名	ANAC_COMP2CR_CMP2EN_Getable
函数原型	FunState ANAC_COMP2CR_CMP2EN_Getable(void)
功能描述	获取比较器 2 使能状态
输入参数 1	无
输出参数	无
返回值	比较器 2 使能状态: DISABLE: 禁止比较器 2 ENABLE: 使能比较器 2

**9.2.82 ANAC\_COMPICR\_CMP2DF\_Setable**

函数名	ANAC_COMPICR_CMP2DF_Setable
函数原型	void ANAC_COMPICR_CMP2DF_Setable(FunState NewState)
功能描述	设置比较器 2 数字滤波使能
输入参数 1	比较器 2 数字滤波使能： DISABLE：禁止比较器 2 数字滤波 ENABLE：使能比较器 2 数字滤波
输出参数	无
返回值	无

**9.2.83 ANAC\_COMPICR\_CMP2DF\_Getable**

函数名	ANAC_COMPICR_CMP2DF_Getable
函数原型	FunState ANAC_COMPICR_CMP2DF_Getable(void)
功能描述	获取比较器 2 数字滤波使能状态
输入参数 1	无
输出参数	无
返回值	比较器 2 数字滤波使能： DISABLE：禁止比较器 2 数字滤波 ENABLE：使能比较器 2 数字滤波

**9.2.84 ANAC\_COMPICR\_CMP2SEL\_Set**

函数名	ANAC_COMPICR_CMP2SEL_Set
函数原型	void ANAC_COMPICR_CMP2SEL_Set(uint32_t SetValue)
功能描述	设置比较器 2 中断源选择
输入参数 1	比较器 2 中断源选择： 00/11：比较器 2 输出上升或下降沿产生中断 01：比较器 2 输出上升沿产生中断 10：比较器 2 输出下降沿产生中断
输出参数	无
返回值	无

**9.2.85 ANAC\_COMPICR\_CMP2SEL\_Get**

函数名	ANAC_COMPICR_CMP2SEL_Get
函数原型	uint32_t ANAC_COMPICR_CMP2SEL_Get(void)

功能描述	获取比较器 2 中断源选择
输入参数 1	无
输出参数	无
返回值	比较器 2 中断源选择： 00/11：比较器 2 输出上升或下降沿产生中断 01：比较器 2 输出上升沿产生中断 10：比较器 2 输出下降沿产生中断

### 9.2.86 ANAC\_COMPICR\_CMP2IE\_Setable

函数名	ANAC_COMPICR_CMP2IE_Setable
函数原型	void ANAC_COMPICR_CMP2IE_Setable(FunState NewState)
功能描述	设置比较器 2 中断使能
输入参数 1	比较器 2 中断使能： DISABLE：禁止比较器 2 中断 ENABLE：使能比较器 2 中断
输出参数	无
返回值	无

### 9.2.87 ANAC\_COMPICR\_CMP2IE\_Getable

函数名	ANAC_COMPICR_CMP2IE_Getable
函数原型	FunState ANAC_COMPICR_CMP2IE_Getable(void)
功能描述	获取比较器 2 中断使能状态
输入参数 1	无
输出参数	无
返回值	比较器 2 中断使能： DISABLE：禁止比较器 2 中断 ENABLE：使能比较器 2 中断

### 9.2.88 ANAC\_COMPIF\_CMP2IF\_Clr

函数名	ANAC_COMPIF_CMP2IF_Clr
函数原型	void ANAC_COMPIF_CMP2IF_Clr(void)
功能描述	清除比较器 2 中断标志
输入参数 1	无
输出参数	无
返回值	无

### 9.2.89 ANAC\_COMPIF\_CMP2IF\_Chk

函数名	ANAC_COMPIF_CMP2IF_Chk
函数原型	FlagStatus ANAC_COMPIF_CMP2IF_Chk(void)
功能描述	获取比较器 2 中断标志状态
输入参数 1	无
输出参数	无
返回值	比较器 2 中断标志状态： 0：未产生比较器 2 中断 1：产生比较器 2 中断标志

### 9.2.90 ANAC\_COMPICR\_BUFBY\_P\_Setable

函数名	ANAC_COMPICR_BUFBY_P_Setable
函数原型	void ANAC_COMPICR_BUFBY_P_Setable(FunState NewState)
功能描述	设置比较器 Buffer Bypass 使能
输入参数 1	比较器 Buffer Bypass 使能： DISABLE：禁止比较器 Buffer Bypass ENABLE：使能比较器 Buffer Bypass
输出参数	无
返回值	无

### 9.2.91 ANAC\_COMPICR\_BUFBY\_P\_Getable

函数名	ANAC_COMPICR_BUFBY_P_Getable
函数原型	FunState ANAC_COMPICR_BUFBY_P_Getable(void)
功能描述	获取比较器 Buffer Bypass 使能状态
输入参数 1	无
输出参数	无
返回值	比较器 Buffer Bypass 使能： DISABLE：禁止比较器 Buffer Bypass ENABLE：使能比较器 Buffer Bypass

### 9.2.92 ANAC\_COMPICR\_BUFENB\_Setable

函数名	ANAC_COMPICR_BUFENB_Setable
函数原型	void ANAC_COMPICR_BUFENB_Setable(FunState NewState)

功能描述	设置比较器 Buffer 使能
输入参数 1	比较器 Buffer 使能: DISABLE: 禁止比较器 Buffer ENABLE: 使能比较器 Buffer
输出参数	无
返回值	无

### 9.2.93 ANAC\_COMPICR\_BUFENB\_Getable

函数名	ANAC_COMPICR_BUFENB_Getable
函数原型	FunState ANAC_COMPICR_BUFENB_Getable(void)
功能描述	获取比较器 Buffer 使能状态
输入参数 1	无
输出参数	无
返回值	比较器 Buffer 使能: DISABLE: 禁止比较器 Buffer ENABLE: 使能比较器 Buffer

## 10 BTIM

### 10.1 BTIM 寄存器结构

寄存器	描述
BT1CR1	BTIM 控制寄存器 1
BT2CR1	
BT1CR2	BTIM 控制寄存器 2
BT2CR2	
BT1CFG1	BTIM 控制设置寄存器 1
BT2CFG1	
BT1CFG2	BTIM 控制设置寄存器 2
BT2CFG2	
BT1PRES	BTIM 预分频寄存器
BT2PRES	
BT1LOADCR	BTIM 加载控制寄存器
BT2LOADCR	
BT1CNTL	BTIM 低位计数器寄存器
BT2CNTL	
BT1CNTH	BTIM 高位计数器寄存器



BT2CNTH	
BT1PRESET	BTIME 预置数寄存器
BT2PRESET	
BT1LOADL	BTIME 低位加载寄存器
BT2LOADL	
BT1LOADH	BTIME 高位加载寄存器
BT2LOADH	
BT1CMPL	BTIME 低位比较寄存器
BT2CMPL	
BT1CMPH	BTIME 高位比较寄存器
BT2CMPH	
BT1OUTCNT	BTIME 输出脉冲宽度寄存器
BT2OUTCNT	
BT1OCR	BTIME 输出控制寄存器
BT2OCR	
BT1IE	BTIME 中断使能寄存器
BT2IE	
BT1IF	BTIME 中断标志寄存器
BT2IF	

## 10.2 BTIM 库函数

序号	函数名	描述
1	BTIMx_Init	BT 初始化
2	BTIMx_Deinit	BT 初始化为缺省值
3	BTIMx_BTCR1_CHEN_Setable	BT 高位计数器启动控制
4	BTIMx_BTCR1_CHEN_Getable	获取 BT 高位计数器启动控制状态
5	BTIMx_BTCR1_CLEN_Setable	BT 低位计数器启动控制
6	BTIMx_BTCR1_CLEN_Getable	获取 BT 低位计数器启动控制状态
7	BTIMx_BTCR1_MODE_Set	设置 BT 工作模式选择
8	BTIMx_BTCR1_MODE_Get	获取 BT 工作模式
9	BTIMx_BTCR1_EDGESEL_Set	设置 BT 计数沿或捕捉沿选择
10	BTIMx_BTCR1_EDGESEL_Get	获取 BT 计数沿或捕捉沿选择
11	BTIMx_BTCR1_CAPMOD_Set	设置 BT 捕捉模式控制
12	BTIMx_BTCR1_CAPMOD_Get	获取 BT 捕捉模式控制
13	BTIMx_BTCR1_CAPCLR_Set	设置 BT 的带清零捕捉模式控制是否



		清零
14	BTIMx_BTCR1_CAPCLR_Get	获取 BT 的带清零捕捉模式控制是否清零状态
15	BTIMx_BTCR1_CAPONCE_Set	设置 BT 是否单次捕捉
16	BTIMx_BTCR1_CAPONCE_Get	获取 BT 是否单次捕捉状态
17	BTIMx_BTCR1_PWM_Setable	设置 BT 的 PWM 输出是否有效
18	BTIMx_BTCR1_PWM_Getable	获取 BT 的 PWM 输出是否有效状态
19	BTIMx_BTCR2_SIG2SEL_Set	设置计数器内部计数源信号选择
20	BTIMx_BTCR2_SIG2SEL_Get	获取计数器内部计数源信号选择
21	BTIMx_BTCR2_SIG1SEL_Set	设置计数器内部捕捉源信号选择
22	BTIMx_BTCR2_SIG1SEL_Get	获取计数器内部捕捉源信号选择
23	BTIMx_BTCR2_CNTHSEL_Set	设置 BT 高位计数器源选择
24	BTIMx_BTCR2_CNTHSEL_Get	获取 BT 高位计数器源选择
25	BTIMx_BTCR2_DIREN_Setable	设置 BT 外部输入 DIR 控制
26	BTIMx_BTCR2_DIREN_Getable	获取 BT 外部输入 DIR 控制状态
27	BTIMx_BTCR2_STDIREN_Setable	设置 BT 内部输入 DIR 控制
28	BTIMx_BTCR2_STDIREN_Getable	获取 BT 内部输入 DIR 控制状态
29	BTIMx_BTCR2_SRCSEL_Set	设置 BT 低位计数器使能控制选择信号
30	BTIMx_BTCR2_SRCSEL_Get	获取 BT 低位计数器使能控制选择信号
31	BTIMx_BTCFG1_RTCSEL1_Set	设置 BTRTCOUT1 信号源
32	BTIMx_BTCFG1_RTCSEL1_Get	获取 BTRTCOUT1 信号源设置
33	BTIMx_BTCFG1_GRP1SEL_Set	设置 BT Group1 信号源
34	BTIMx_BTCFG1_GRP1SEL_Get	获取 BT Group1 信号源设置
35	BTIMx_BTCFG2_INSEL1_Get	设置 BT 内部输入信号源 1
36	BTIMx_BTCFG2_INSEL1_Get	获取 BT 内部输入信号源 1 设置
37	BTIMx_BTCFG2_EXSEL1_Set	设置 BT 外部输入信号源 1
38	BTIMx_BTCFG2_EXSEL1_Get	获取 BT 外部输入信号源 1 设置
39	BTIMx_BTCFG1_RTCSEL2_Set	设置 BTRTCOUT2 信号源
40	BTIMx_BTCFG1_RTCSEL2_Get	获取 BTRTCOUT2 信号源设置
41	BTIMx_BTCFG1_GRP2SEL_Set	设置 BT Group2 信号源
42	BTIMx_BTCFG1_GRP2SEL_Get	获取 BT Group2 信号源设置
43	BTIMx_BTCFG2_EXSEL2_Set	设置 BT 外部输入信号源 2
44	BTIMx_BTCFG2_EXSEL2_Get	获取 BT 外部输入信号源 2 设置
45	BTIMx_BTCFG2_INSEL2_Set	设置 BT 内部输入信号源 2
46	BTIMx_BTCFG2_INSEL2_Get	获取 BT 内部输入信号源 2 设置



47	BTIMx_BTCR2_DIRPO_Set	设置 BT 输入信号 2 极性
48	BTIMx_BTCR2_DIRPO_Get	获取 BT 输入信号 2 极性
49	BTIMx_BTPRES_Write	设置 BT Group1 预分频
50	BTIMx_BTPRES_Read	获取 BT Group1 预分频
51	BTIMx_BTLOADCR_LHEN_Setable	使能 BT 高位加载
52	BTIMx_BTLOADCR_LLEN_Setable	使能 BT 低位加载
53	BTIMx_BTCNTL_Read	读取计数器低位计数值
54	BTIMx_BTCNTH_Read	读取计数器高位计数值
55	BTIMx_BTPRESET_PRESETH_Set	设置计数器高位预置值
56	BTIMx_BTPRESET_PRESETH_Get	获取计数器高位预置值
57	BTIMx_BTPRESET_PRESETL_Set	设置计数器低位预置值
58	BTIMx_BTPRESET_PRESETL_Get	获取计数器低位预置值
59	BTIMx_BTLOADL_Write	设置计数器低位加载数
60	BTIMx_BTLOADL_Read	读取计数器低位加载数
61	BTIMx_BTLOADH_Write	设置计数器高位加载数
62	BTIMx_BTLOADH_Read	读取计数器高位加载数
63	BTIMx_BTCMPL_Write	设置计数器低位比较值
64	BTIMx_BTCMPL_Read	读取计数器低位比较值
65	BTIMx_BTCMPH_Write	设置计数器高位比较值
66	BTIMx_BTCMPH_Read	读取计数器高位比较值
67	BTIMx_BTOUTCNT_Write	设置计数器具体脉冲宽度
68	BTIMx_BTOUTCNT_Read	读取计数器具体脉冲宽度
69	BTIMx_BTOCR_OUTCLR_Set	设置计数器输出是否清零
70	BTIMx_BTOCR_OUTCLR_Get	获取计数器输出是否清零
71	BTIMx_BTOCR_OUTINV_Set	设置计数器输出电平
72	BTIMx_BTOCR_OUTINV_Get	获取计数器输出电平
73	BTIMx_BTOCR_OUTMOD_Set	设置计数器输出模式
74	BTIMx_BTOCR_OUTMOD_Get	获取计数器输出模式
75	BTIMx_BTOCR_OUTSEL_Set	设置计数器输出信号源
76	BTIMx_BTOCR_OUTSEL_Get	获取计数器输出信号源
77	BTIMx_BTIE_CMPHIE_Setable	使能 BT 高位比较发生信号
78	BTIMx_BTIE_CMPHIE_Getable	获取 BT 高位比较发生信号状态
79	BTIMx_BTIE_CMPLIE_Setable	使能 BT 低位比较发生信号
80	BTIMx_BTIE_CMPLIE_Getable	获取 BT 低位比较发生信号状态
81	BTIMx_BTIE_OVHIE_Setable	使能 BT 高位溢出信号
82	BTIMx_BTIE_OVHIE_Getable	获取 BT 高位溢出信号状态
83	BTIMx_BTIE_OVLIE_Setable	使能 BT 低位溢出信号



84	BTIMx_BTIE_OVLIE_Getable	获取 BT 低位溢出信号状态
85	BTIMx_BTIE_CAPIE_Setable	使能 BT 捕捉产生信号
86	BTIMx_BTIE_CAPIE_Getable	获取 BT 捕捉产生信号状态
87	BTIMx_BTIF_EDGESTA_Clr	清除计数器捕捉沿状态
88	BTIMx_BTIF_EDGESTA_Chk	获取计数器捕捉沿状态
89	BTIMx_BTIF_CMPHIF_Clr	清除计数器高位比较发生信号状态
90	BTIMx_BTIF_CMPHIF_Chk	获取计数器高位比较发生信号状态
91	BTIMx_BTIF_CMPLIF_Clr	清除计数器低位比较发生信号状态
92	BTIMx_BTIF_CMPLIF_Chk	获取计数器低位比较发生信号状态
93	BTIMx_BTIF_OVHIF_Clr	清除计数器高位溢出发生信号状态
94	BTIMx_BTIF_OVHIF_Chk	获取计数器高位溢出发生信号状态
95	BTIMx_BTIF_OVLIF_Clr	清除计数器低位溢出发生信号状态
96	BTIMx_BTIF_OVLIF_Chk	获取计数器低位溢出发生信号状态
97	BTIMx_BTIF_CAPIF_Clr	清除计数器捕捉发生信号状态
98	BTIMx_BTIF_CAPIF_Chk	获取计数器捕捉发生信号状态

### 10.2.1 BTIMx\_Init

函数名	BTIMx_Init
函数原型	void BTIMx_Init(BTIMx_Type* BTIMx, BTIM_InitTypeDef* para)
功能描述	BT 初始化
输入参数 1	定时器名称：BTIM1/BTIM2
输入参数 2	<div>定时器参数：<div><div>控制类参数：<div><div>分频系数</div><div>高位预加载值</div><div>低位预加载值</div><div>高位加载值</div><div>低位加载值</div></div><div>工作模式：<div>0：8 位定时/计数模式</div><div>1：16 位捕捉模式</div></div><div>跳变沿选择：<div>0：上升沿</div><div>1：下降沿</div></div></div></div><div>信号来源参数：<div><div>RTCOUT1 信号选择控制 1：</div><div>00：32768Hz，XTLF 时钟输出</div><div>01：RTCSec，由 RTC 模块输出的秒信号</div></div></div></div>



	10: RTCMin, 由 RTC 模块输出的分信号 11: LPTIM_OUT, 由 LPTIM 模块输出的信号	
	RTCOUT2 信号选择控制 2: 00: 32768Hz, XTLF 时钟输出 01: RTCSec, 由 RTC 模块输出的秒信号 10: RTCMin, 由 RTC 模块输出的分信号 11: LPTIM_OUT, 由 LPTIM 模块输出的信号	
	内部输入信号选择控制 1: 00: UART_RX0/ UART_RX0 01: UART_RX1/ UART_RX1 10: UART_RX2/ UART_RX2 11: PCLP/PCLP	
	内部输入信号选择控制 2: 00: UART_RX3/ UART_RX3 01: UART_RX4/ UART_RX4 10: UART_RX5/ UART_RX5 11: PCLP/BT1_OUT	
	外部输入信号选择控制 1 00: BT1_IN0/BT2_IN0 01: BT1_IN1/BT2_IN1 10: BT1_IN2/BT2_IN2 11: BT1_IN3/BT2_IN3	
	外部输入信号选择控制 2 00: BT1_IN0/BT2_IN0 01: BT1_IN1/BT2_IN1 10: BT1_IN2/BT2_IN2 11: BT1_IN3/BT2_IN3	
	Group1 信号选择控制: 00: APKCLK 01: RTCOUT1 10: IN_SIG1 11: EX_SIG1	
	Group2 信号选择控制: 00: APKCLK 01: RTCOUT2 10: IN_SIG2 11: EX_SIG2	
	内部捕获源信号选择: 0: 内部捕获源信号选择 Group2 1: 内部捕获源信号选择 Group1	
	内部计数源信号选择: 0: 内部计数源信号选择 Group1 1: 内部计数源信号选择 Group2	
	低位计数器计数使能控制选择信号:	



	<p>0: 低位计数器计数使能端选则由寄存器 STDIR 或外部 EX_SIG2 输入控制</p> <p>1: 低位计数器计数使能端选则常使能, 此时低位计数器计数不受 DIR 控制</p> <p>高 8 位计数器计数源选择</p> <p>00/11: 选择 ET1 的低位计数器的溢出信号, 与低位计数器组成 16 位计数器</p> <p>01: 选择内部捕捉源信号</p> <p>10: 选择内部计数源信号或外部 DIR 输入组合信号</p> <p>捕捉模式参数:</p> <p>单次捕捉控制:</p> <p>0: 连续捕捉</p> <p>1: 单次捕捉有效</p> <p>带清零捕捉模式控制:</p> <p>0: 捕捉不清零</p> <p>1: 捕捉清零</p> <p>捕捉模式控制:</p> <p>0: 脉冲周期捕捉</p> <p>1: 脉冲宽度捕捉</p> <p>正反脉冲计数参数:</p> <p>外部输入 DIR 控制使能:</p> <p>0: 外部输入 DIR 信号无效</p> <p>1: 外部输入 DIR 信号有效</p> <p>输入信号 2 极性选择:</p> <p>0: 对外部输入 DIR 信号 EX_SIG2 不反向</p> <p>1: 对外部输入 DIR 信号 EX_SIG2 反向</p> <p>输出参数:</p> <p>计数器输出脉冲宽度计数器: 调整脉冲宽度</p> <p>输出清零, 该位硬件自动清零, 清零后输出由原信号驱动:</p> <p>0: 写入无效</p> <p>1: 输出清零, 若设置了输出反响则清零后输出为高电平, 反之为低电平。</p> <p>输出点平反向选择:</p> <p>0: 输出点平反向</p> <p>1: 输出点平反向</p> <p>输出模式选择:</p> <p>0: 输出脉冲, 脉冲宽度可调</p> <p>1: 输出之前的反向脉冲</p> <p>输出信号选择:</p> <p>000: 输出高位计数器的比较信号, 只在计数模式有效</p> <p>001: 输出低位计数器的比较信号, 只在计数模式有效</p> <p>010: 直接输出 Group1 的输入信号, 计数模式、捕捉模式有效</p> <p>011: 直接输出 Group2 的输入信号, 计数模式、捕捉模式有效</p> <p>其他: PWM 输出</p>	
输出参数	无	
返回值	无	



### 10.2.2 BTIMx\_Deinit

函数名	BTIMx_Deinit
函数原型	void BTIMx_Deinit(BTIMx_Type* BTIMx)
功能描述	BT 初始化为默认值
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	无

### 10.2.3 BTIMx\_BTCR1\_CHEN\_Setable

函数名	BTIMx_BTCR1_CHEN_Setable
函数原型	void BTIMx_BTCR1_CHEN_Setable(BTIMx_Type* BTIMx, FunState NewState)
功能描述	BT 高位计数器启动控制
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 高位计数器启动控制: DISABLE : 停止高位计数器 ENABLE : 启动高位计数器
输出参数	无
返回值	无

### 10.2.4 BTIMx\_BTCR1\_CHEN\_Getable

函数名	BTIMx_BTCR1_CHEN_Getable
函数原型	FunState BTIMx_BTCR1_CHEN_Getable(BTIMx_Type* BTIMx)
功能描述	获取 BT 高位计数器启动控制状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 高位计数器启动控制: DISABLE: 停止当前 BT 高位计数器 ENABLE: 启动当前 BT 高位计数器

### 10.2.5 BTIMx\_BTCR1\_CLEN\_Setable

函数名	BTIMx_BTCR1_CLEN_Setable
函数原型	void BTIMx_BTCR1_CLEN_Setable(BTIMx_Type* BTIMx, FunState NewState)
功能描述	BT 低位计数器启动控制
输入参数 1	定时器名称: BTIM1/BTIM2

输入参数 2	BT 低位计数器启动控制： DISABLE：停止低位计数器 ENABLE：启动低位计数器
输出参数	无
返回值	无

### 10.2.6 BTIMx\_BTCR1\_CLEN\_Getable

函数名	BTIMx_BTCR1_CLEN_Getable
函数原型	FunState BTIMx_BTCR1_CLEN_Getable(BTIMx_Type* BTIMx)
功能描述	获取 BT 低位计数器启动控制状态
输入参数 1	定时器名称：BTIM1/BTIM2
输出参数	无
返回值	BT 低位计数器启动控制： DISABLE：停止低位计数器 ENABLE：启动低位计数器

### 10.2.7 BTIMx\_BTCR1\_MODE\_Set

函数名	BTIMx_BTCR1_MODE_Set
函数原型	void BTIMx_BTCR1_MODE_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	设置 BT 工作模式选择
输入参数 1	定时器名称：BTIM1/BTIM2
输入参数 2	BT 工作模式： BTIMx_BTCR1_MODE_8BITS_TIM_CNT：8 位定时器 BTIMx_BTCR1_MODE_16BITS_CAP：16 位定时器
输出参数	无
返回值	无

### 10.2.8 BTIMx\_BTCR1\_MODE\_Get

函数名	BTIMx_BTCR1_MODE_Get
函数原型	uint32_t BTIMx_BTCR1_MODE_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 工作模式
输入参数 1	定时器名称：BTIM1/BTIM2
输出参数	无
返回值	BT 工作模式： BTIMx_BTCR1_MODE_8BITS_TIM_CNT：8 位定时器



BTIMx\_BTCR1\_MODE\_16BITS\_CAP: 16 位定时器

### 10.2.9 BTIMx\_BTCR1\_EDGESEL\_Set

函数名	BTIMx_BTCR1_EDGESEL_Set
函数原型	void BTIMx_BTCR1_EDGESEL_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	设置 BT 计数沿或捕捉沿选择
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 计数沿或捕捉沿选择: BTIMx_BTCR1_EDGESEL_POS: 计数沿或捕捉沿为上升沿 BTIMx_BTCR1_EDGESEL_NEG: 计数沿或捕捉沿为下降沿
输出参数	无
返回值	无

### 10.2.10 BTIMx\_BTCR1\_EDGESEL\_Get

函数名	BTIMx_BTCR1_EDGESEL_Get
函数原型	uint32_t BTIMx_BTCR1_EDGESEL_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 计数沿或捕捉沿选择
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 计数沿或捕捉沿选择: BTIMx_BTCR1_EDGESEL_POS: 计数沿或捕捉沿为上升沿 BTIMx_BTCR1_EDGESEL_NEG: 计数沿或捕捉沿为下降沿

### 10.2.11 BTIMx\_BTCR1\_CAPMOD\_Set

函数名	BTIMx_BTCR1_CAPMOD_Set
函数原型	void BTIMx_BTCR1_CAPMOD_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT 捕捉模式控制
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 捕捉模式控制: BTIMx_BTCR1_CAPMOD_PAUSE_PERIOD: 脉冲周期捕捉 BTIMx_BTCR1_CAPMOD_PAUSE_WIDTH: 脉冲宽度捕捉
输出参数	无
返回值	无

**10.2.12 BTIMx\_BTCR1\_CAPMOD\_Get**

函数名	BTIMx_BTCR1_CAPMOD_Get
函数原型	uint32_t BTIMx_BTCR1_CAPMOD_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 捕捉模式控制
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 捕捉模式控制: BTIMx_BTCR1_CAPMOD_PAUSE_PERIOD: 脉冲周期捕捉 BTIMx_BTCR1_CAPMOD_PAUSE_WIDTH: 脉冲宽度捕捉

**10.2.13 BTIMx\_BTCR1\_CAPCLR\_Set**

函数名	BTIMx_BTCR1_CAPCLR_Set
函数原型	void BTIMx_BTCR1_CAPCLR_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT 的带清零捕捉模式控制是否清零
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 的带清零捕捉模式控制是否清零: BTIMx_BTCR1_CAPCLR_CAP_CNT_CLR: 不论在脉冲宽度还是周期捕捉情况下, 捕捉到第一个沿后将计数器清零产生中断, 捕捉到第二个沿后锁存 (锁存到高地位预置数寄存器), 计数值并同时清零计数器。 BTIMx_BTCR1_CAPCLR_CAP_CNT_NO_CLR: 捕捉后计数器不清零, 计数器一直自由计数
输出参数	无
返回值	无

**10.2.14 BTIMx\_BTCR1\_CAPCLR\_Get**

函数名	BTIMx_BTCR1_CAPCLR_Get
函数原型	uint32_t BTIMx_BTCR1_CAPCLR_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 的带清零捕捉模式控制是否清零状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 的带清零捕捉模式控制是否清零: BTIMx_BTCR1_CAPCLR_CAP_CNT_CLR: 不论在脉冲宽度还是周期捕捉情况下, 捕捉到第一个沿后将计数器清零产生中断, 捕捉到第二个沿后锁存 (锁存到高地位预置数寄存器), 计数值并同时清零计数器。 BTIMx_BTCR1_CAPCLR_CAP_CNT_NO_CLR: 捕捉后计数器不清零, 计数器一直自由

计数

**10.2.15 BTIMx\_BTCR1\_CAPONCE\_Set**

函数名	BTIMx_BTCR1_CAPONCE_Set
函数原型	void BTIMx_BTCR1_CAPONCE_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT 是否单次捕捉
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 是否单次捕捉: BTIMx_BTCR1_CAPONCE_CONTINUE: 连续捕捉 BTIMx_BTCR1_CAPONCE_SINGLE: 单次捕捉
输出参数	无
返回值	无

**10.2.16 BTIMx\_BTCR1\_CAPONCE\_Get**

函数名	BTIMx_BTCR1_CAPONCE_Get
函数原型	uint32_t BTIMx_BTCR1_CAPONCE_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 是否单次捕捉状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 是否单次捕捉: BTIMx_BTCR1_CAPONCE_CONTINUE: 连续捕捉 BTIMx_BTCR1_CAPONCE_SINGLE: 单次捕捉

**10.2.17 BTIMx\_BTCR1\_PWM\_Setable**

函数名	BTIMx_BTCR1_PWM_Setable
函数原型	void BTIMx_BTCR1_PWM_Setable(BTIMx_Type* BTIMx, FunState NewState)
功能描述	设置 BT 的 PWM 输出是否有效
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 的 PWM 输出是否有效: DISABLE: PWM 输出模式无效 ENABLE: PWM 输出模式有效
输出参数	无
返回值	无

**10.2.18 BTIMx\_BTCR1\_PWM\_Getable**

函数名	BTIMx_BTCR1_PWM_Getable
函数原型	FunState BTIMx_BTCR1_PWM_Getable(BTIMx_Type* BTIMx)
功能描述	获取 BT 的 PWM 输出是否有效状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 的 PWM 输出是否有效: DISABLE: PWM 输出模式无效 ENABLE: PWM 输出模式有效

**10.2.19 BTIMx\_BTCR2\_SIG2SEL\_Set**

函数名	BTIMx_BTCR2_SIG2SEL_Set
函数原型	void BTIMx_BTCR2_SIG2SEL_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器内部计数源信号选择
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	计数器内部计数源信号选择: BTIMx_BTCR2_SIG2SEL_GROUP1: 内部计数源选择 Group1 BTIMx_BTCR2_SIG2SEL_GROUP2: 内部计数源选择 Group2
输出参数	无
返回值	无

**10.2.20 BTIMx\_BTCR2\_SIG2SEL\_Get**

函数名	BTIMx_BTCR2_SIG2SEL_Get
函数原型	uint32_t BTIMx_BTCR2_SIG2SEL_Get(BTIMx_Type* BTIMx)
功能描述	获取计数器内部计数源信号选择
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器内部计数源信号选择: BTIMx_BTCR2_SIG2SEL_GROUP1: 内部计数源选择 Group1 BTIMx_BTCR2_SIG2SEL_GROUP2: 内部计数源选择 Group2

**10.2.21 BTIMx\_BTCR2\_SIG1SEL\_Set**

函数名	BTIMx_BTCR2_SIG1SEL_Set
函数原型	void BTIMx_BTCR2_SIG1SEL_Set(BTIMx_Type* BTIMx, uint32_t SetValue)

功能描述	配置计数器内部捕捉源信号选择
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	计数器内部捕捉源信号选择: BTIMx_BTCR2_SIG1SEL_GROUP1: 内部捕捉源选择 Group1 BTIMx_BTCR2_SIG1SEL_GROUP2: 内部捕捉源选择 Group2
输出参数	无
返回值	无

### 10.2.22 BTIMx\_BTCR2\_SIG1SEL\_Get

函数名	BTIMx_BTCR2_SIG1SEL_Get
函数原型	uint32_t BTIMx_BTCR2_SIG1SEL_Get(BTIMx_Type* BTIMx)
功能描述	获取计数器内部捕捉源信号选择
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器内部捕捉源信号选择: BTIMx_BTCR2_SIG1SEL_GROUP1: 内部捕捉源选择 Group1 BTIMx_BTCR2_SIG1SEL_GROUP2: 内部捕捉源选择 Group2

### 10.2.23 BTIMx\_BTCR2\_CNTHSEL\_Set

函数名	BTIMx_BTCR2_CNTHSEL_Set
函数原型	void BTIMx_BTCR2_CNTHSEL_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT 高位计数器源选择
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 高位计数器源选择: BTIMx_BTCR2_CNTHSEL_CNTL: 与低位计数器组成 16 位计数器 BTIMx_BTCR2_CNTHSEL_CAPSRC: 选择内部捕捉信号源 BTIMx_BTCR2_CNTHSEL_CNTRSRC_DIR: 选择内部计数信号源或外部 DIR 输入组合
输出参数	无
返回值	无

### 10.2.24 BTIMx\_BTCR2\_CNTHSEL\_Get

函数名	BTIMx_BTCR2_CNTHSEL_Get
函数原型	uint32_t BTIMx_BTCR2_CNTHSEL_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 高位计数器源选择
输入参数 1	定时器名称: BTIM1/BTIM2



输出参数	无
返回值	BT 高位计数器源选择： BTIMx_BTCR2_CNTHSEL_CNTL：与低位计数器组成 16 位计数器 BTIMx_BTCR2_CNTHSEL_CAPSRC：选择内部捕捉信号源 BTIMx_BTCR2_CNTHSEL_CNTSRC_DIR：选择内部计数信号源或外部 DIR 输入组合

### 10.2.25 BTIMx\_BTCR2\_DIREN\_Setable

函数名	BTIMx_BTCR2_DIREN_Setable
函数原型	void BTIMx_BTCR2_DIREN_Setable(BTIMx_Type* BTIMx, FunState NewState)
功能描述	配置 BT 外部输入 DIR 控制
输入参数 1	定时器名称：BTIM1/BTIM2
输入参数 2	BT 外部输入 DIR 控制： DISABLE：外部输入 DIR 控制无效 ENABLE：外部输入 DIR 控制有效
输出参数	无
返回值	无

### 10.2.26 BTIMx\_BTCR2\_DIREN\_Getable

函数名	BTIMx_BTCR2_DIREN_Getable
函数原型	FunState BTIMx_BTCR2_DIREN_Getable(BTIMx_Type* BTIMx)
功能描述	获取 BT 外部输入 DIR 控制状态
输入参数 1	定时器名称：BTIM1/BTIM2
输出参数	无
返回值	BT 外部输入 DIR 控制： DISABLE：外部输入 DIR 控制无效 ENABLE：外部输入 DIR 控制有效

### 10.2.27 BTIMx\_BTCR2\_STDIREN\_Setable

函数名	BTIMx_BTCR2_STDIREN_Setable
函数原型	void BTIMx_BTCR2_STDIREN_Setable(BTIMx_Type* BTIMx, FunState NewState)
功能描述	配置 BT 内部输入 DIR 控制
输入参数 1	定时器名称：BTIM1/BTIM2
输入参数 2	BT 内部输入 DIR 控制： DISABLE：内部输入 DIR 控制无效 ENABLE：内部输入 DIR 控制有效

输出参数	无
返回值	无

### 10.2.28 BTIMx\_BTCR2\_STDIREnable

函数名	BTIMx_BTCR2_STDIREnable
函数原型	FunState BTIMx_BTCR2_STDIREnable(BTIMx_Type* BTIMx)
功能描述	获取 BT 内部输入 DIR 控制状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 内部输入 DIR 控制: DISABLE: 内部输入 DIR 控制无效 ENABLE: 内部输入 DIR 控制有效

### 10.2.29 BTIMx\_BTCR2\_SRCSEL\_Set

函数名	BTIMx_BTCR2_SRCSEL_Set
函数原型	void BTIMx_BTCR2_SRCSEL_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT 低位计数器使能控制选择信号
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 低位计数器使能控制选择信号: BTIMx_BTCR2_SRCSEL_WITHOUT_DIR: 低位计数器不受 DIR 控制 BTIMx_BTCR2_SRCSEL_STDIREnable: 低位计数器由 STDIREnable 或外部 EX_SIG2 输入控制
输出参数	无
返回值	无

### 10.2.30 BTIMx\_BTCR2\_SRCSEL\_Get

函数名	BTIMx_BTCR2_SRCSEL_Get
函数原型	uint32_t BTIMx_BTCR2_SRCSEL_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 低位计数器使能控制选择信号
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 低位计数器使能控制选择信号: BTIMx_BTCR2_SRCSEL_WITHOUT_DIR: 低位计数器不受 DIR 控制 BTIMx_BTCR2_SRCSEL_STDIREnable: 低位计数器由 STDIREnable 或外部 EX_SIG2 输入控制



### 10.2.31 BTIMx\_BTCFG1\_RTCSEL1\_Set

函数名	BTIMx_BTCFG1_RTCSEL1_Set
函数原型	void BTIMx_BTCFG1_RTCSEL1_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT RTCOUT1 信号源
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT RTCOUT1 信号源: BTIMx_BTCFG1_RTCSEL1_RTC_32768: RTC 模块输出的 32768Hz 信号 BTIMx_BTCFG1_RTCSEL1_RTC_SEC: RTC 模块输出的秒信号 BTIMx_BTCFG1_RTCSEL1_RTC_MIN: RTC 模块输出的分信号 BTIMx_BTCFG1_RTCSEL1_LPTIM_OUT: LPTIM 模块输出的信号
输出参数	无
返回值	无

### 10.2.32 BTIMx\_BTCFG1\_RTCSEL1\_Get

函数名	BTIMx_BTCFG1_RTCSEL1_Get
函数原型	uint32_t BTIMx_BTCFG1_RTCSEL1_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT RTCOUT1 信号源配置
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT RTCOUT1 信号源: BTIMx_BTCFG1_RTCSEL1_RTC_32768: RTC 模块输出的 32768Hz 信号 BTIMx_BTCFG1_RTCSEL1_RTC_SEC: RTC 模块输出的秒信号 BTIMx_BTCFG1_RTCSEL1_RTC_MIN: RTC 模块输出的分信号 BTIMx_BTCFG1_RTCSEL1_LPTIM_OUT: LPTIM 模块输出的信号

### 10.2.33 BTIMx\_BTCFG1\_GRP1SEL\_Set

函数名	BTIMx_BTCFG1_GRP1SEL_Set
函数原型	void BTIMx_BTCFG1_GRP1SEL_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT Group1 信号源
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT Group1 信号源: BTIMx_BTCFG1_GRP1SEL_APBCLK: 系统时钟 APBCLK BTIMx_BTCFG1_GRP1SEL_RTCOUT2: RTC 时钟 RTCOUT2 BTIMx_BTCFG1_GRP1SEL_IN_SIG2: 内部输入信号 2



	BTIMx_BTCFG1_GRP1SEL_EX_SIG2: 外部输入信号 2
输出参数	无
返回值	无

### 10.2.34 BTIMx\_BTCFG1\_GRP1SEL\_Get

函数名	BTIMx_BTCFG1_GRP1SEL_Get
函数原型	uint32_t BTIMx_BTCFG1_GRP1SEL_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT Group1 信号源配置
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT Group1 信号源: BTIMx_BTCFG1_GRP1SEL_APBCLK: 系统时钟 APBCLK BTIMx_BTCFG1_GRP1SEL_RTCOUT2: RTC 时钟 RTCOUT2 BTIMx_BTCFG1_GRP1SEL_IN_SIG2: 内部输入信号 2 BTIMx_BTCFG1_GRP1SEL_EX_SIG2: 外部输入信号 2

### 10.2.35 BTIMx\_BTCFG2\_INSEL1\_Get

函数名	BTIMx_BTCFG2_INSEL1_Get
函数原型	uint32_t BTIMx_BTCFG2_INSEL1_Get(BTIMx_Type* BTIMx)
功能描述	配置 BT 内部输入信号源 1
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 内部输入信号源 1: BTIMx_BTCFG2_INSEL1_UART_RX0: 信号源为 UART0 接收信号 BTIMx_BTCFG2_INSEL1_UART_RX1: 信号源为 UART1 接收信号 BTIMx_BTCFG2_INSEL1_UART_RX2: 信号源为 UART2 接收信号 BTIMx_BTCFG2_INSEL1_RCLP: 信号源为 RCLP 做信号源

### 10.2.36 BTIMx\_BTCFG2\_INSEL1\_Get

函数名	BTIMx_BTCFG2_INSEL1_Get
函数原型	uint32_t BTIMx_BTCFG2_INSEL1_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 内部输入信号源 1 配置
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 内部输入信号源 1:



	BTIMx_BTCFG2_INSEL1_UART_RX0: 信号源为 UART0 接收信号 BTIMx_BTCFG2_INSEL1_UART_RX1: 信号源为 UART1 接收信号 BTIMx_BTCFG2_INSEL1_UART_RX2: 信号源为 UART2 接收信号 BTIMx_BTCFG2_INSEL1_RCLP: 信号源为 RCLP 做信号源
--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 10.2.37 BTIMx\_BTCFG2\_EXSEL1\_Set

函数名	BTIMx_BTCFG2_EXSEL1_Set
函数原型	void BTIMx_BTCFG2_EXSEL1_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT 外部输入信号源 1
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 外部输入信号源 1: BTIMx_BTCFG2_EXSEL1_BT_IN0: 外部输入 1 BTIMx_BTCFG2_EXSEL1_BT_IN1: 外部输入 2 BTIMx_BTCFG2_EXSEL1_BT_IN2: 外部输入 3 BTIMx_BTCFG2_EXSEL1_BT_IN3: 外部输入 4
输出参数	无
返回值	无

### 10.2.38 BTIMx\_BTCFG2\_EXSEL1\_Get

函数名	BTIMx_BTCFG2_EXSEL1_Get
函数原型	uint32_t BTIMx_BTCFG2_EXSEL1_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 外部输入信号源 1 配置
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 外部输入信号源 1: BTIMx_BTCFG2_EXSEL1_BT_IN0: 外部输入 1 BTIMx_BTCFG2_EXSEL1_BT_IN1: 外部输入 2 BTIMx_BTCFG2_EXSEL1_BT_IN2: 外部输入 3 BTIMx_BTCFG2_EXSEL1_BT_IN3: 外部输入 4

### 10.2.39 BTIMx\_BTCFG1\_RTCSEL2\_Set

函数名	BTIMx_BTCFG1_RTCSEL2_Set
函数原型	void BTIMx_BTCFG1_RTCSEL2_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT RTCOUT2 信号源
输入参数 1	定时器名称: BTIM1/BTIM2



输入参数 2	BT RTCOUT2 信号源: BTIMx_BTCFG1_RTCSEL2_RTC_32768: RTC 模块输出的 32768Hz 信号 BTIMx_BTCFG1_RTCSEL2_RTC_SEC: RTC 模块输出的秒信号 BTIMx_BTCFG1_RTCSEL2_RTC_MIN: RTC 模块输出的分信号 BTIMx_BTCFG1_RTCSEL2_LPTIM_OUT: LPTIM 模块输出的信号
输出参数	无
返回值	无

#### 10.2.40 BTIMx\_BTCFG1\_RTCSEL2\_Get

函数名	BTIMx_BTCFG1_RTCSEL2_Get
函数原型	uint32_t BTIMx_BTCFG1_RTCSEL2_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT RTCOUT2 信号源配置
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT RTCOUT2 信号源: BTIMx_BTCFG1_RTCSEL2_RTC_32768: RTC 模块输出的 32768Hz 信号 BTIMx_BTCFG1_RTCSEL2_RTC_SEC: RTC 模块输出的秒信号 BTIMx_BTCFG1_RTCSEL2_RTC_MIN: RTC 模块输出的分信号 BTIMx_BTCFG1_RTCSEL2_LPTIM_OUT: LPTIM 模块输出的信号

#### 10.2.41 BTIMx\_BTCFG1\_GRP2SEL\_Set

函数名	BTIMx_BTCFG1_GRP2SEL_Set
函数原型	void BTIMx_BTCFG1_GRP2SEL_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT Group2 信号源
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT Group2 信号源: BTIMx_BTCFG1_GRP2SEL_APBCLK : 系统时钟 APBCLK BTIMx_BTCFG1_GRP2SEL_RTCOUT2: RTC 时钟 RTCOUT2 BTIMx_BTCFG1_GRP2SEL_IN_SIG2: 内部输入信号 2 BTIMx_BTCFG1_GRP2SEL_EX_SIG2: 外部输入信号 2
输出参数	无
返回值	无

#### 10.2.42 BTIMx\_BTCFG1\_GRP2SEL\_Get

函数名	BTIMx_BTCFG1_GRP2SEL_Get
-----	--------------------------

函数原型	uint32_t BTIMx_BTCFG1_GRP2SEL_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT Group2 信号源配置
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT Group2 信号源: BTIMx_BTCFG1_GRP2SEL_APBCLK : 系统时钟 APBCLK BTIMx_BTCFG1_GRP2SEL_RTCOUT2: RTC 时钟 RTCOUT2 BTIMx_BTCFG1_GRP2SEL_IN_SIG2: 内部输入信号 2 BTIMx_BTCFG1_GRP2SEL_EX_SIG2: 外部输入信号 2

### 10.2.43 BTIMx\_BTCFG2\_EXSEL2\_Set

函数名	BTIMx_BTCFG2_EXSEL2_Set
函数原型	void BTIMx_BTCFG2_EXSEL2_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT 外部输入信号源 2
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 外部输入信号源 2: BTIMx_BTCFG2_EXSEL2_BT_IN0: 外部输入 1 BTIMx_BTCFG2_EXSEL2_BT_IN1: 外部输入 2 BTIMx_BTCFG2_EXSEL2_BT_IN2: 外部输入 3 BTIMx_BTCFG2_EXSEL2_BT_IN3: 外部输入 4
输出参数	无
返回值	无

### 10.2.44 BTIMx\_BTCFG2\_EXSEL2\_Get

函数名	BTIMx_BTCFG2_EXSEL2_Get
函数原型	uint32_t BTIMx_BTCFG2_EXSEL2_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 外部输入信号源 2 配置
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 外部输入信号源 2: BTIMx_BTCFG2_EXSEL2_BT_IN0: 外部输入 1 BTIMx_BTCFG2_EXSEL2_BT_IN1: 外部输入 2 BTIMx_BTCFG2_EXSEL2_BT_IN2: 外部输入 3 BTIMx_BTCFG2_EXSEL2_BT_IN3: 外部输入 4

**10.2.45 BTIMx\_BTCFG2\_INSEL2\_Set**

函数名	BTIMx_BTCFG2_INSEL2_Set
函数原型	void BTIMx_BTCFG2_INSEL2_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT 内部输入信号源 2
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 内部输入信号源 2: BTIMx_BTCFG2_INSEL2_UART_RX3: 选择 UART3 接收信号做信号源 BTIMx_BTCFG2_INSEL2_UART_RX4: 选择 UART4 接收信号做信号源 BTIMx_BTCFG2_INSEL2_UART_RX5: 选择 UART5 接收信号做信号源 BTIMx_BTCFG2_INSEL2_RCLP: 选择 RCLP 做信号源 BTIMx_BTCFG2_INSEL2_BT1_OUT: 选择 BT1 的输出做信号源
输出参数	无
返回值	无

**10.2.46 BTIMx\_BTCFG2\_INSEL2\_Get**

函数名	BTIMx_BTCFG2_INSEL2_Get
函数原型	uint32_t BTIMx_BTCFG2_INSEL2_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 内部输入信号源 2 配置
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 内部输入信号源 2: BTIMx_BTCFG2_INSEL2_UART_RX3: 选择 UART3 接收信号做信号源 BTIMx_BTCFG2_INSEL2_UART_RX4: 选择 UART4 接收信号做信号源 BTIMx_BTCFG2_INSEL2_UART_RX5: 选择 UART5 接收信号做信号源 BTIMx_BTCFG2_INSEL2_RCLP: 选择 RCLP 做信号源 BTIMx_BTCFG2_INSEL2_BT1_OUT: 选择 BT1 的输出做信号源

**10.2.47 BTIMx\_BTCR2\_DIRPO\_Set**

函数名	BTIMx_BTCR2_DIRPO_Set
函数原型	void BTIMx_BTCR2_DIRPO_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT 输入信号 2 极性
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 输入信号 2 极性: BTIMx_BTCR2_DIRPO_ANTI: 对外部输入 DIR 信号取反向 BTIMx_BTCR2_DIRPO_NO_ANTI: 对外部输入 DIR 信号不反向

输出参数	无
返回值	无

#### 10.2.48 BTIMx\_BTCR2\_DIRPO\_Get

函数名	BTIMx_BTCR2_DIRPO_Get
函数原型	uint32_t BTIMx_BTCR2_DIRPO_Get(BTIMx_Type* BTIMx)
功能描述	获取 BT 输入信号 2 极性
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 输入信号 2 极性: BTIMx_BTCR2_DIRPO_ANTI: 对外部输入 DIR 信号取反向 BTIMx_BTCR2_DIRPO_NO_ANTI: 对外部输入 DIR 信号不反向

#### 10.2.49 BTIMx\_BTPRES\_Write

函数名	BTIMx_BTPRES_Write
函数原型	void BTIMx_BTPRES_Write(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置 BT Group1 预分频
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	具体分频值: 00-FF, 00 表示 1 分频, FF 表示 256 分频
输出参数	无
返回值	无

#### 10.2.50 BTIMx\_BTPRES\_Read

函数名	BTIMx_BTPRES_Read
函数原型	uint32_t BTIMx_BTPRES_Read(BTIMx_Type* BTIMx)
功能描述	获取 BT Group1 预分频
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	具体分频值: 00-FF 00 表示 1 分频, FF 表示 256 分频

#### 10.2.51 BTIMx\_BTLOADCR\_LHEN\_Setable

函数名	BTIMx_BTLOADCR_LHEN_Setable
函数原型	void BTIMx_BTLOADCR_LHEN_Setable(BTIMx_Type* BTIMx, FunState NewState)
功能描述	使能 BT 高位加载

输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 高位加载使能: DISABLE: 禁止高位加载 ENABLE: 使能高位加载
输出参数	无
返回值	无

### 10.2.52 BTIMx\_BTLOADCR\_LLEN\_Setable

函数名	BTIMx_BTLOADCR_LLEN_Setable
函数原型	void BTIMx_BTLOADCR_LLEN_Setable(BTIMx_Type* BTIMx, FunState NewState)
功能描述	使能 BT 低位加载
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 高位加载使能: DISABLE: 禁止高位加载 ENABLE: 使能高位加载
输出参数	无
返回值	无

### 10.2.53 BTIMx\_BTCNTL\_Read

函数名	BTIMx_BTCNTL_Read
函数原型	uint32_t BTIMx_BTCNTL_Read(BTIMx_Type* BTIMx)
功能描述	读取计数器低位计数值
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器低位计数值: 0000-FFFF

### 10.2.54 BTIMx\_BTCNTH\_Read

函数名	BTIMx_BTCNTH_Read
函数原型	uint32_t BTIMx_BTCNTH_Read(BTIMx_Type* BTIMx)
功能描述	读取计数器低位计数值
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器低位计数值: 0000-FFFF



**10.2.55 BTIMx\_BTPRESET\_PRESETH\_Set**

函数名	BTIMx_BTPRESET_PRESETH_Set
函数原型	void BTIMx_BTPRESET_PRESETH_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器高位预置值
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	计数器高位预置值: 0000-FFFF
输出参数	无
返回值	无

**10.2.56 BTIMx\_BTPRESET\_PRESETH\_Get**

函数名	BTIMx_BTPRESET_PRESETH_Get
函数原型	uint32_t BTIMx_BTPRESET_PRESETH_Get(BTIMx_Type* BTIMx)
功能描述	获取计数器高位预置值
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器高位预置值: 0000-FFFF

**10.2.57 BTIMx\_BTPRESET\_PRESETL\_Set**

函数名	BTIMx_BTPRESET_PRESETL_Set
函数原型	void BTIMx_BTPRESET_PRESETL_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器低位预置值
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	计数器低位预置值: 0000-FFFF
输出参数	无
返回值	无

**10.2.58 BTIMx\_BTPRESET\_PRESETL\_Get**

函数名	BTIMx_BTPRESET_PRESETL_Get
函数原型	uint32_t BTIMx_BTPRESET_PRESETL_Get(BTIMx_Type* BTIMx)
功能描述	获取计数器低位预置值
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器低位预置值: 0000-FFFF



**10.2.59 BTIMx\_BTLOADL\_Write**

函数名	BTIMx_BTLOADL_Write
函数原型	void BTIMx_BTLOADL_Write(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器低位加载数
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	计数器低位加载数: 0000-FFFF
输出参数	无
返回值	无

**10.2.60 BTIMx\_BTLOADL\_Read**

函数名	BTIMx_BTLOADL_Read
函数原型	uint32_t BTIMx_BTLOADL_Read(BTIMx_Type* BTIMx)
功能描述	读取计数器低位加载数
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器低位加载数: 0000-FFFF

**10.2.61 BTIMx\_BTLOADH\_Write**

函数名	BTIMx_BTLOADH_Write
函数原型	void BTIMx_BTLOADH_Write(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器高位加载数
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	计数器高位加载数: 0000-FFFF
输出参数	无
返回值	无

**10.2.62 BTIMx\_BTLOADH\_Read**

函数名	BTIMx_BTLOADH_Read
函数原型	uint32_t BTIMx_BTLOADH_Read(BTIMx_Type* BTIMx)
功能描述	读取计数器高位加载数
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器高位加载数: 0000-FFFF

**10.2.63 BTIMx\_BTCMPL\_Write**

函数名	BTIMx_BTCMPL_Write
函数原型	void BTIMx_BTCMPL_Write(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器低位比较值
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	计数器低位比较值: 0000-FFFF
输出参数	无
返回值	无

**10.2.64 BTIMx\_BTCMPL\_Read**

函数名	BTIMx_BTCMPL_Read
函数原型	uint32_t BTIMx_BTCMPL_Read(BTIMx_Type* BTIMx)
功能描述	读取计数器低位比较值
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器低位比较值: 0000-FFFF

**10.2.65 BTIMx\_BTCMPH\_Write**

函数名	BTIMx_BTCMPH_Write
函数原型	void BTIMx_BTCMPH_Write(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器高位比较值
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	计数器高位比较值: 0000-FFFF
输出参数	无
返回值	无

**10.2.66 BTIMx\_BTCMPH\_Read**

函数名	BTIMx_BTCMPH_Read
函数原型	uint32_t BTIMx_BTCMPH_Read(BTIMx_Type* BTIMx)
功能描述	读取计数器高位比较值
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器高位比较值: 0000-FFFF

**10.2.67 BTIMx\_BTOUTCNT\_Write**

函数名	BTIMx_BTOUTCNT_Write
函数原型	void BTIMx_BTOUTCNT_Write(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器具体脉冲宽度
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	计数器具体脉冲宽度: 0000-0FFF      宽度为 $30.5 \times (1 + \text{SetValue})$
输出参数	无
返回值	无

**10.2.68 BTIMx\_BTOUTCNT\_Read**

函数名	BTIMx_BTOUTCNT_Read
函数原型	uint32_t BTIMx_BTOUTCNT_Read(BTIMx_Type* BTIMx)
功能描述	读取计数器具体脉冲宽度
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器具体脉冲宽度: 0000-0FFF      宽度为 $30.5 \times (1 + \text{SetValue})$

**10.2.69 BTIMx\_BTOCR\_OUTCLR\_Set**

函数名	BTIMx_BTOCR_OUTCLR_Set
函数原型	void BTIMx_BTOCR_OUTCLR_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器输出是否清零
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	计数器输出是否清零: BTIMx_BTOCR_OUTCLR_CLR: 输出清零(若设置了反向则输出高, 未设置则输出低) BTIMx_BTOCR_OUTCLR: 输出写入无效
输出参数	无
返回值	无

**10.2.70 BTIMx\_BTOCR\_OUTCLR\_Get**

函数名	BTIMx_BTOCR_OUTCLR_Get
函数原型	uint32_t BTIMx_BTOCR_OUTCLR_Get(BTIMx_Type* BTIMx)
功能描述	获取计数器输出是否清零
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无

返回值	计数器输出是否清零： BTIMx_BTOCR_OUTCLR_CLR：输出清零(若设置了反向则输出高，未设置则输出低) BTIMx_BTOCR_OUTCLR：输出写入无效
-----	--------------------------------------------------------------------------------------------

### 10.2.71 BTIMx\_BTOCR\_OUTINV\_Set

函数名	BTIMx_BTOCR_OUTINV_Set
函数原型	void BTIMx_BTOCR_OUTINV_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器输出电平
输入参数 1	定时器名称：BTIM1/BTIM2
输入参数 2	计数器输出电平： BTIMx_BTOCR_OUTINV_ANTI：输出电平取反 BTIMx_BTOCR_OUTINV_NOT_ANTI：输出电平不取反
输出参数	无
返回值	无

### 10.2.72 BTIMx\_BTOCR\_OUTINV\_Get

函数名	BTIMx_BTOCR_OUTINV_Get
函数原型	uint32_t BTIMx_BTOCR_OUTINV_Get(BTIMx_Type* BTIMx)
功能描述	获取计数器输出电平
输入参数 1	定时器名称：BTIM1/BTIM2
输出参数	无
返回值	计数器输出电平： BTIMx_BTOCR_OUTINV_ANTI：输出电平取反 BTIMx_BTOCR_OUTINV_NOT_ANTI：输出电平不取反

### 10.2.73 BTIMx\_BTOCR\_OUTMOD\_Set

函数名	BTIMx_BTOCR_OUTMOD_Set
函数原型	void BTIMx_BTOCR_OUTMOD_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器输出模式
输入参数 1	定时器名称：BTIM1/BTIM2
输入参数 2	计数器输出模式： BTIMx_BTOCR_OUTINV_ANTI：输出之前的反向电平 BTIMx_BTOCR_OUTINV_NOT_ANTI：输出脉冲，宽度可调
输出参数	无
返回值	无

### 10.2.74 BTIMx\_BTOCR\_OUTMOD\_Get

函数名	BTIMx_BTOCR_OUTMOD_Get
函数原型	uint32_t BTIMx_BTOCR_OUTMOD_Get(BTIMx_Type* BTIMx)
功能描述	获取计数器输出模式
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器输出模式: BTIMx_BTOCR_OUTINV_ANTI: 输出之前的反向电平 BTIMx_BTOCR_OUTINV_NOT_ANTI: 输出脉冲, 宽度可调

### 10.2.75 BTIMx\_BTOCR\_OUTSEL\_Set

函数名	BTIMx_BTOCR_OUTSEL_Set
函数原型	void BTIMx_BTOCR_OUTSEL_Set(BTIMx_Type* BTIMx, uint32_t SetValue)
功能描述	配置计数器输出信号源
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	计数器输出信号源: BTIMx_BTOCR_OUTSEL_CMPH: 输出高位计数器比较信号, 仅计数模式有效 BTIMx_BTOCR_OUTSEL_CMPL: 输出低位计数器比较信号, 仅计数模式有效 BTIMx_BTOCR_OUTSEL_GROUP1: 输出 Group1 的输入信号, 计数, 捕捉模式有效 BTIMx_BTOCR_OUTSEL_GROUP2: 输出 Group2 的输入信号, 计数, 捕捉模式有效 BTIMx_BTOCR_OUTSEL_PWM: PWM 输出
输出参数	无
返回值	无

### 10.2.76 BTIMx\_BTOCR\_OUTSEL\_Get

函数名	BTIMx_BTOCR_OUTSEL_Get
函数原型	uint32_t BTIMx_BTOCR_OUTSEL_Get(BTIMx_Type* BTIMx)
功能描述	获取计数器输出信号源
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器输出信号源: BTIMx_BTOCR_OUTSEL_CMPH: 输出高位计数器比较信号, 仅计数模式有效 BTIMx_BTOCR_OUTSEL_CMPL: 输出低位计数器比较信号, 仅计数模式有效 BTIMx_BTOCR_OUTSEL_GROUP1: 输出 Group1 的输入信号, 计数, 捕捉模式有效



BTIMx\_BTOCR\_OUTSEL\_GROUP2: 输出 Group2 的输入信号, 计数, 捕捉模式有效  
BTIMx\_BTOCR\_OUTSEL\_PWM: PWM 输出

### 10.2.77 BTIMx\_BTIE\_CMPHIE\_Setable

函数名	BTIMx_BTIE_CMPHIE_Setable
函数原型	void BTIMx_BTIE_CMPHIE_Setable(BTIMx_Type* BTIMx, FunState NewState)
功能描述	使能 BT 高位比较发生信号
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 高位比较发生信号使能: DISABLE: 禁止高位比较发生信号中断 ENABLE: 使能高位比较发生信号中断
输出参数	无
返回值	无

### 10.2.78 BTIMx\_BTIE\_CMPHIE\_Getable

函数名	BTIMx_BTIE_CMPHIE_Getable
函数原型	FunState BTIMx_BTIE_CMPHIE_Getable(BTIMx_Type* BTIMx)
功能描述	获取 BT 高位比较发生信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 高位比较发生信号使能: DISABLE: 禁止高位比较发生信号中断 ENABLE: 使能高位比较发生信号中断

### 10.2.79 BTIMx\_BTIE\_CMPLIE\_Setable

函数名	BTIMx_BTIE_CMPLIE_Setable
函数原型	void BTIMx_BTIE_CMPLIE_Setable(BTIMx_Type* BTIMx, FunState NewState)
功能描述	使能 BT 低位比较发生信号
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 低位比较发生信号使能: DISABLE: 禁止低位比较发生信号中断 ENABLE: 使能低位比较发生信号中断
输出参数	无
返回值	无

**10.2.80 BTIMx\_BTIE\_CMPLIE\_Getable**

函数名	BTIMx_BTIE_CMPLIE_Getable
函数原型	FunState BTIMx_BTIE_CMPLIE_Getable(BTIMx_Type* BTIMx)
功能描述	获取 BT 低位比较发生信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 低位比较发生信号使能: DISABLE: 禁止低位比较发生信号中断 ENABLE: 使能低位比较发生信号中断

**10.2.81 BTIMx\_BTIE\_OVHIE\_Setable**

函数名	BTIMx_BTIE_OVHIE_Setable
函数原型	void BTIMx_BTIE_OVHIE_Setable(BTIMx_Type* BTIMx, FunState NewState)
功能描述	使能 BT 高位溢出信号
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 高位溢出信号使能: DISABLE: 禁止高位溢出发生信号中断、 ENABLE: 使能高位溢出发生信号中断
输出参数	无
返回值	无

**10.2.82 BTIMx\_BTIE\_OVHIE\_Getable**

函数名	BTIMx_BTIE_OVHIE_Getable
函数原型	FunState BTIMx_BTIE_OVHIE_Getable(BTIMx_Type* BTIMx)
功能描述	获取 BT 高位溢出信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 高位溢出信号使能: DISABLE: 禁止高位溢出发生信号中断、 ENABLE: 使能高位溢出发生信号中断

**10.2.83 BTIMx\_BTIE\_OVLIE\_Setable**

函数名	BTIMx_BTIE_OVLIE_Setable
函数原型	void BTIMx_BTIE_OVLIE_Setable(BTIMx_Type* BTIMx, FunState NewState)



功能描述	使能 BT 低位溢出信号
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 低位溢出信号使能: DISABLE: 禁止低位溢出发生信号中断 ENABLE: 使能低位溢出发生信号中断
输出参数	无
返回值	无

#### 10.2.84 BTIMx\_BTIE\_OVLIE\_Getable

函数名	BTIMx_BTIE_OVLIE_Getable
函数原型	FunState BTIMx_BTIE_OVLIE_Getable(BTIMx_Type* BTIMx)
功能描述	获取 BT 低位溢出信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	BT 低位溢出信号使能: DISABLE: 禁止低位溢出发生信号中断 ENABLE: 使能低位溢出发生信号中断

#### 10.2.85 BTIMx\_BTIE\_CAPIE\_Setable

函数名	BTIMx_BTIE_CAPIE_Setable
函数原型	void BTIMx_BTIE_CAPIE_Setable(BTIMx_Type* BTIMx, FunState NewState)
功能描述	使能 BT 捕捉产生信号
输入参数 1	定时器名称: BTIM1/BTIM2
输入参数 2	BT 捕捉产生信号使能: DISABLE: 禁止捕捉发生信号中断 ENABLE: 使能捕捉发生信号中断
输出参数	无
返回值	无

#### 10.2.86 BTIMx\_BTIE\_CAPIE\_Getable

函数名	BTIMx_BTIE_CAPIE_Getable
函数原型	FunState BTIMx_BTIE_CAPIE_Getable(BTIMx_Type* BTIMx)
功能描述	获取 BT 捕捉产生信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无



返回值	BT 捕捉产生信号使能： DISABLE：禁止捕捉发生信号中断 ENABLE：使能捕捉发生信号中断
-----	---------------------------------------------------------

### 10.2.87 BTIMx\_BTIF\_EDGESTA\_Clr

函数名	BTIMx_BTIF_EDGESTA_Clr
函数原型	void BTIMx_BTIF_EDGESTA_Clr(BTIMx_Type* BTIMx)
功能描述	清除计数器捕捉沿状态
输入参数 1	定时器名称：BTIM1/BTIM2
输出参数	无
返回值	无

### 10.2.88 BTIMx\_BTIF\_EDGESTA\_Chk

函数名	BTIMx_BTIF_EDGESTA_Chk
函数原型	FlagStatus BTIMx_BTIF_EDGESTA_Chk(BTIMx_Type* BTIMx)
功能描述	获取计数器捕捉沿状态
输入参数 1	定时器名称：BTIM1/BTIM2
输出参数	无
返回值	计数器捕捉沿状态： RESET：脉冲宽度捕捉模式时表示捕捉到上沿 SET：脉冲宽度捕捉模式时表示捕捉到下沿

### 10.2.89 BTIMx\_BTIF\_CMPHIF\_Clr

函数名	BTIMx_BTIF_CMPHIF_Clr
函数原型	void BTIMx_BTIF_CMPHIF_Clr(BTIMx_Type* BTIMx)
功能描述	清除计数器高位比较发生信号状态
输入参数 1	定时器名称：BTIM1/BTIM2
输出参数	无
返回值	无

### 10.2.90 BTIMx\_BTIF\_CMPHIF\_Chk

函数名	BTIMx_BTIF_CMPHIF_Chk
函数原型	FlagStatus BTIMx_BTIF_CMPHIF_Chk(BTIMx_Type* BTIMx)
功能描述	获取计数器高位比较发生信号状态

输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器高位比较发生信号状态: RESET: 当前计数器的值小于工作寄存器的值 SET: 当前计数器的值大于等工作寄存器的值

### 10.2.91 BTIMx\_BTIF\_CMPLIF\_Clr

函数名	BTIMx_BTIF_CMPLIF_Clr
函数原型	void BTIMx_BTIF_CMPLIF_Clr(BTIMx_Type* BTIMx)
功能描述	清除计数器低位比较发生信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	无

### 10.2.92 BTIMx\_BTIF\_CMPLIF\_Chk

函数名	BTIMx_BTIF_CMPLIF_Chk
函数原型	FlagStatus BTIMx_BTIF_CMPLIF_Chk(BTIMx_Type* BTIMx)
功能描述	获取计数器低位比较发生信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器低位比较发生信号状态: RESET: 当前计数器的值小于工作寄存器的值 SET: 当前计数器的值大于等工作寄存器的值

### 10.2.93 BTIMx\_BTIF\_OVHIF\_Clr

函数名	BTIMx_BTIF_OVHIF_Clr
函数原型	void BTIMx_BTIF_OVHIF_Clr(BTIMx_Type* BTIMx)
功能描述	清除计数器高位溢出发生信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	无

### 10.2.94 BTIMx\_BTIF\_OVHIF\_Chk

函数名	BTIMx_BTIF_OVHIF_Chk
-----	----------------------

函数原型	FlagStatus BTIMx_BTIF_OVHIF_Chk(BTIMx_Type* BTIMx)
功能描述	获取计数器高位溢出发生信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器高位溢出发生信号状态: RESET: 未产生溢出 SET: 产生计数溢出

### 10.2.95 BTIMx\_BTIF\_OVLIF\_Clr

函数名	BTIMx_BTIF_OVLIF_Clr
函数原型	void BTIMx_BTIF_OVLIF_Clr(BTIMx_Type* BTIMx)
功能描述	清除计数器低位溢出发生信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	无

### 10.2.96 BTIMx\_BTIF\_OVLIF\_Chk

函数名	BTIMx_BTIF_OVLIF_Chk
函数原型	FlagStatus BTIMx_BTIF_OVLIF_Chk(BTIMx_Type* BTIMx)
功能描述	获取计数器低位溢出发生信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器低位溢出发生信号状态: RESET: 未产生溢出 SET: 产生计数溢出

### 10.2.97 BTIMx\_BTIF\_CAPIF\_Clr

函数名	BTIMx_BTIF_CAPIF_Clr
函数原型	void BTIMx_BTIF_CAPIF_Clr(BTIMx_Type* BTIMx)
功能描述	清除计数器捕捉发生信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	无

### 10.2.98 BTIMx\_BTIF\_CAPIF\_Chk

函数名	BTIMx_BTIF_CAPIF_Chk
函数原型	FlagStatus BTIMx_BTIF_CAPIF_Chk(BTIMx_Type* BTIMx)
功能描述	获取计数器捕捉发生信号状态
输入参数 1	定时器名称: BTIM1/BTIM2
输出参数	无
返回值	计数器捕捉发生信号状态: RESET: 未捕捉到指定的沿 SET: 捕捉到指定的沿

# 11 LPTIM

## 11.1 LPTIM 寄存器结构

寄存器	描述
LPTCFG	LPTIM 设置寄存器
LPTCNT	LPTIM 计数寄存器
LPTCMP	LPTIM 比较值寄存器
LPTTARGET	LPTIM 目标值寄存器
LPTIE	LPTIM 中断使能寄存器
LPTIF	LPTIM 中断标志寄存器
LPTCTRL	LPTIM 控制寄存器

## 11.2 定时器中断库函数

序号	函数名	描述
1	LPTIM_Init	LPTIM 初始化
2	LPTIM_Deinit	LPTIM 初始化为缺省值
3	LPTIM_LPTCTRL_LPTEN_Setable	设置 LPTIM 使能
4	LPTIM_LPTCTRL_LPTEN_Getable	获取 LPTIM 使能状态
5	LPTIM_LPTCFG_CLKSEL_Set	设置计时时钟源选择
6	LPTIM_LPTCFG_CLKSEL_Get	获取计时时钟源选择
7	LPTIM_LPTCFG_DIVSEL_Set	设置时钟分频选择
8	LPTIM_LPTCFG_DIVSEL_Get	获取时钟分频选择
9	LPTIM_LPTCFG_TMODE_Set	设置 LPTIM 工作模式选择
10	LPTIM_LPTCFG_TMODE_Get	获取 LPTIM 工作模式选择
11	LPTIM_LPTCNT_Write	写入计数值
12	LPTIM_LPTCNT_Read	读取计数值
13	LPTIM_LPTCMP_Write	写入比较值
14	LPTIM_LPTCMP_Read	读取比较值
15	LPTIM_LPTTARGET_Write	写入目标值
16	LPTIM_LPTTARGET_Read	读取目标值
17	LPTIM_LPTCFG_MODE_Set	设置计数模式选择
18	LPTIM_LPTCFG_MODE_Get	获取计数模式选择
19	LPTIM_LPTCFG_EDGESEL_Set	设置输入边沿选择

20	LPTIM_LPTCFG_EDGESEL_Get	获取输入边沿选择
21	LPTIM_LPTCFG_TRIGCFG_Set	设置外部触发沿选择
22	LPTIM_LPTCFG_TRIGCFG_Get	获取外部触发沿选择
23	LPTIM_LPTCFG_POLARITY_Set	设置输出波形极性选择
24	LPTIM_LPTCFG_POLARITY_Get	获取输出波形极性选择
25	LPTIM_LPTIE_OVIE_Setable	设置计数器溢出中断
26	LPTIM_LPTIE_OVIE_Getable	获取计数器溢出中断状态
27	LPTIM_LPTIE_COMPIE_Setable	设置比较值匹配中断
28	LPTIM_LPTIE_COMPIE_Getable	获取比较值匹配中断状态
29	LPTIM_LPTIE_TRIGIE_Setable	设置外部触发中断
30	LPTIM_LPTIE_TRIGIE_Getable	获取外部触发中断状态
31	LPTIM_LPTIF_OVIF_Clr	清除计数器溢出标志
32	LPTIM_LPTIF_OVIF_Chk	获取计数器溢出标志状态
33	LPTIM_LPTIF_COMPIF_Clr	清除比较值匹配标志
34	LPTIM_LPTIF_COMPIF_Chk	获取比较值匹配标志状态
35	LPTIM_LPTIF_TRIGIF_Clr	清除外部触发标志
36	LPTIM_LPTIF_TRIGIF_Chk	获取外部触发标志状态
37	LPTIM_LPTCFG_FLTEN_Setable	设置外部 Trigger 滤波使能
38	LPTIM_LPTCFG_FLTEN_Getable	获取外部 Trigger 滤波使能状态
39	LPTIM_LPTCFG_PWM_Set	设置 PWM 模式选择
40	LPTIM_LPTCFG_PWM_Get	获取 PWM 模式选择

### 11.2.1 LPTIM\_Init

函数名	LPTIM_Init
函数原型	void LPTIM_Init(LPTIM_InitTypeDef* para)
功能描述	LPTIM 初始化
输入参数 1	<div>LPIM 参数:</div> <div><div>工作模式选择: 00: 带波形输出的普通定时器模式 01: Trigger 脉冲触发计数模式 10: 外部异步脉冲计数模式 11: Timeout 模式</div><div>计数模式选择: 0: 连续计数模式 1: 单次计数模式</div><div>脉宽调制模式: 0: 周期方波输出模式 1: PWM 输出模式</div></div>

	时钟极性选择： 0：正极性波形，即第一次计数值=比较值时产生输出波形上升沿 1：负极性波形，即第一次计数值=比较值时产生输出波形下降沿
	外部触发边沿选择： 00：外部输入信号上升沿 trigger 01：外部输入信号下降沿 trigger 10/11：外部输入信号上升下降沿 trigger
	外部 Trigger 滤波使能： DISABLE：禁止外部 Trigger 滤波 ENABLE：使能外部 Trigger 滤波
	LPTIN 输入边沿选择 0：LPTIN 的上升沿计数 1：LPTIN 的下降沿计数
	时钟源选择： 00：LSCLK 作为计数时钟 01：RCLP 作为计数时钟 10：PCLK 的门控时钟作为计数时钟 11：LPTIN 作为计数时钟
	时钟分频选择： 000：1 分频 001：2 分频 010：4 分频 011：8 分频 100：16 分频 101：32 分频 110：64 分频 111：128 分频
	比较值： 0000-FFFF
	目标值： 0000-FFFF
输出参数	无
返回值	无

### 11.2.2 LPTIM\_Deinit

函数名	LPTIM_Deinit
函数原型	void LPTIM_Deinit(void)
功能描述	LPTIM 初始化为默认值
输入参数 1	无
输出参数	无
返回值	无

### 11.2.3 LPTIM\_LPTCTRL\_LPTEN\_Setable

函数名	LPTIM_LPTCTRL_LPTEN_Setable
函数原型	void LPTIM_LPTCTRL_LPTEN_Setable(FunState NewState)
功能描述	设置 LPTIM 使能
输入参数 1	LPTIM 计数使能: DISABLE: 禁止计数器计数 ENABLE: 使能计数器计数
输出参数	无
返回值	无

### 11.2.4 LPTIM\_LPTCTRL\_LPTEN\_Getable

函数名	LPTIM_LPTCTRL_LPTEN_Getable
函数原型	FunState LPTIM_LPTCTRL_LPTEN_Getable(void)
功能描述	获取 LPTIM 使能状态
输入参数 1	无
输出参数	无
返回值	LPTIM 计数使能: DISABLE: 禁止计数器计数 ENABLE: 使能计数器计数

### 11.2.5 LPTIM\_LPTCFG\_CLKSEL\_Set

函数名	LPTIM_LPTCFG_CLKSEL_Set
函数原型	void LPTIM_LPTCFG_CLKSEL_Set(uint32_t SetValue)
功能描述	设置计时时钟源选择
输入参数 1	时钟源: 00: LSCLK 作为计数时钟 01: RCLP 作为计数时钟 10: PCLK 的门控时钟作为计数时钟 11: LPTIN 作为计数时钟
输出参数	无
返回值	无



### 11.2.6 LPTIM\_LPTCFG\_CLKSEL\_Get

函数名	LPTIM_LPTCFG_CLKSEL_Get
函数原型	uint32_t LPTIM_LPTCFG_CLKSEL_Get(void)
功能描述	获取计时时钟源选择
输入参数 1	无
输出参数	无
返回值	时钟源： 00: LSCLK 作为计数时钟 01: RCLP 作为计数时钟 10: PCLK 的门控时钟作为计数时钟 11: LPTIN 作为计数时钟

### 11.2.7 LPTIM\_LPTCFG\_DIVSEL\_Set

函数名	LPTIM_LPTCFG_DIVSEL_Set
函数原型	void LPTIM_LPTCFG_DIVSEL_Set(uint32_t SetValue)
功能描述	设置时钟分频选择
输入参数 1	分频数值： 000: 1 分频 001: 2 分频 010: 4 分频 011: 8 分频 100: 16 分频 101: 32 分频 110: 64 分频 111: 128 分频
输出参数	无
返回值	无

### 11.2.8 LPTIM\_LPTCFG\_DIVSEL\_Get

函数名	LPTIM_LPTCFG_DIVSEL_Get
函数原型	uint32_t LPTIM_LPTCFG_DIVSEL_Get(void)
功能描述	获取时钟分频选择
输入参数 1	无
输出参数	无
返回值	分频数值：

	000: 1 分频
	001: 2 分频
	010: 4 分频
	011: 8 分频
	100: 16 分频
	101: 32 分频
	110: 64 分频
	111: 128 分频

### 11.2.9 LPTIM\_LPTCFG\_TMODE\_Set

函数名	LPTIM_LPTCFG_TMODE_Set
函数原型	void LPTIM_LPTCFG_TMODE_Set(uint32_t SetValue)
功能描述	设置 LPTIM 工作模式选择
输入参数 1	工作模式选择: 00: 带波形输出的普通定时器模式 01: Trigger 脉冲触发计数模式 10: 外部异步脉冲计数模式 11: Timeout 模式
输出参数	无
返回值	无

### 11.2.10 LPTIM\_LPTCFG\_TMODE\_Get

函数名	LPTIM_LPTCFG_TMODE_Get
函数原型	uint32_t LPTIM_LPTCFG_TMODE_Get(void)
功能描述	获取 LPTIM 工作模式选择
输入参数 1	无
输出参数	无
返回值	工作模式选择: 00: 带波形输出的普通定时器模式 01: Trigger 脉冲触发计数模式 10: 外部异步脉冲计数模式 11: Timeout 模式

### 11.2.11 LPTIM\_LPTCNT\_Write

函数名	LPTIM_LPTCNT_Write
-----	--------------------

函数原型	void LPTIM_LPTCNT_Write(uint32_t SetValue)
功能描述	写入计数值
输入参数 1	计数值: 0000-FFFF
输出参数	无
返回值	无

### 11.2.12 LPTIM\_LPTCNT\_Read

函数名	LPTIM_LPTCNT_Read
函数原型	uint32_t LPTIM_LPTCNT_Read(void)
功能描述	读取计数值
输入参数 1	无
输出参数	无
返回值	计数值: 0000-FFFF

### 11.2.13 LPTIM\_LPTCMP\_Write

函数名	LPTIM_LPTCMP_Write
函数原型	void LPTIM_LPTCMP_Write(uint32_t SetValue)
功能描述	写入比较值
输入参数 1	比较值: 0000-FFFF
输出参数	无
返回值	无

### 11.2.14 LPTIM\_LPTCMP\_Read

函数名	LPTIM_LPTCMP_Read
函数原型	uint32_t LPTIM_LPTCMP_Read(void)
功能描述	读取比较值
输入参数 1	无
输出参数	无
返回值	比较值: 0000-FFFF

### 11.2.15 LPTIM\_LPTTARGET\_Write

函数名	LPTIM_LPTTARGET_Write
函数原型	void LPTIM_LPTTARGET_Write(uint32_t SetValue)
功能描述	写入目标值

输入参数 1	目标值: 0000-FFFF
输出参数	无
返回值	无

### 11.2.16 LPTIM\_LPTTARGET\_Read

函数名	LPTIM_LPTTARGET_Read
函数原型	uint32_t LPTIM_LPTTARGET_Read(void)
功能描述	读取目标值
输入参数 1	无
输出参数	无
返回值	目标值: 0000-FFFF

### 11.2.17 LPTIM\_LPTCFG\_MODE\_Set

函数名	LPTIM_LPTCFG_MODE_Set
函数原型	void LPTIM_LPTCFG_MODE_Set(uint32_t SetValue)
功能描述	设置计数模式选择
输入参数 1	计数模式选择 0: 连续计数模式 1: 单次计数模式
输出参数	无
返回值	无

### 11.2.18 LPTIM\_LPTCFG\_MODE\_Get

函数名	LPTIM_LPTCFG_MODE_Get
函数原型	uint32_t LPTIM_LPTCFG_MODE_Get(void)
功能描述	获取计数模式选择
输入参数 1	无
输出参数	无
返回值	计数模式选择 0: 连续计数模式 1: 单次计数模式

### 11.2.19 LPTIM\_LPTCFG\_EDGESEL\_Set

函数名	LPTIM_LPTCFG_EDGESEL_Set
-----	--------------------------

函数原型	void LPTIM_LPTCFG_EDGESEL_Set(uint32_t SetValue)
功能描述	设置输入边沿选择
输入参数 1	输入边沿选择 0: LPTIN 的上升沿计数 1: LPTIN 的下降沿计数
输出参数	无
返回值	无

### 11.2.20 LPTIM\_LPTCFG\_EDGESEL\_Get

函数名	LPTIM_LPTCFG_EDGESEL_Get
函数原型	uint32_t LPTIM_LPTCFG_EDGESEL_Get(void)
功能描述	获取输入边沿选择
输入参数 1	无
输出参数	无
返回值	输入边沿选择 0: LPTIN 的上升沿计数 1: LPTIN 的下降沿计数

### 11.2.21 LPTIM\_LPTCFG\_TRIGCFG\_Set

函数名	LPTIM_LPTCFG_TRIGCFG_Set
函数原型	void LPTIM_LPTCFG_TRIGCFG_Set(uint32_t SetValue)
功能描述	设置外部触发沿选择
输入参数 1	外部触发边沿选择: 00: 外部输入信号上升沿 trigger 01: 外部输入信号下降沿 trigger 10/11: 外部输入信号上升下降沿 trigger
输出参数	无
返回值	无

### 11.2.22 LPTIM\_LPTCFG\_TRIGCFG\_Get

函数名	LPTIM_LPTCFG_TRIGCFG_Get
函数原型	uint32_t LPTIM_LPTCFG_TRIGCFG_Get(void)
功能描述	获取外部触发沿选择
输入参数 1	无
输出参数	无

返回值	外部触发边沿选择： 00：外部输入信号上升沿 trigger 01：外部输入信号下降沿 trigger 10/11：外部输入信号上升下降沿 trigger
-----	----------------------------------------------------------------------------------------

### 11.2.23 LPTIM\_LPTCFG\_POLARITY\_Set

函数名	LPTIM_LPTCFG_POLARITY_Set
函数原型	void LPTIM_LPTCFG_POLARITY_Set(uint32_t SetValue)
功能描述	设置输出波形极性选择
输入参数 1	输出波形极性选择： 0：正极性波形，即第一次计数值=比较值时产生输出波形上升沿 1：负极性波形，即第一次计数值=比较值时产生输出波形下降沿
输出参数	无
返回值	无

### 11.2.24 LPTIM\_LPTCFG\_POLARITY\_Get

函数名	LPTIM_LPTCFG_POLARITY_Get
函数原型	uint32_t LPTIM_LPTCFG_POLARITY_Get(void)
功能描述	获取输出波形极性选择
输入参数 1	无
输出参数	无
返回值	输出波形极性选择： 0：正极性波形，即第一次计数值=比较值时产生输出波形上升沿 1：负极性波形，即第一次计数值=比较值时产生输出波形下降沿

### 11.2.25 LPTIM\_LPTIE\_OVIE\_Setable

函数名	LPTIM_LPTIE_OVIE_Setable
函数原型	void LPTIM_LPTIE_OVIE_Setable(FunState NewState)
功能描述	设置计数器溢出中断使能
输入参数 1	计数器溢出中断： DISABLE：禁止计数器溢出中断 ENABLE：使能计数器溢出中断
输出参数	无
返回值	无

**11.2.26 LPTIM\_LPTIE\_OVIE\_Getable**

函数名	LPTIM_LPTIE_OVIE_Getable
函数原型	FunState LPTIM_LPTIE_OVIE_Getable(void)
功能描述	获取计数器溢出中断使能状态
输入参数 1	无
输出参数	无
返回值	计数器溢出中断: DISABLE: 禁止计数器溢出中断 ENABLE: 使能计数器溢出中断

**11.2.27 LPTIM\_LPTIE\_COMPIE\_Setable**

函数名	LPTIM_LPTIE_COMPIE_Setable
函数原型	void LPTIM_LPTIE_COMPIE_Setable(FunState NewState)
功能描述	设置比较值匹配中断使能
输入参数 1	比较值匹配中断: DISABLE: 禁止计数器值和比较值匹配中断 ENABLE: 使能计数器值和比较值匹配中断
输出参数	无
返回值	无

**11.2.28 LPTIM\_LPTIE\_COMPIE\_Getable**

函数名	LPTIM_LPTIE_COMPIE_Getable
函数原型	FunState LPTIM_LPTIE_COMPIE_Getable(void)
功能描述	获取比较值匹配中断状态
输入参数 1	无
输出参数	无
返回值	比较值匹配中断: DISABLE: 禁止计数器值和比较值匹配中断 ENABLE: 使能计数器值和比较值匹配中断

**11.2.29 LPTIM\_LPTIE\_TRIGIE\_Setable**

函数名	LPTIM_LPTIE_TRIGIE_Setable
函数原型	void LPTIM_LPTIE_TRIGIE_Setable(FunState NewState)
功能描述	设置外部触发中断使能

输入参数 1	外部触发中断： DISABLE：禁止外部触发中断 ENABLE：使能外部触发中断
输出参数	无
返回值	无

### 11.2.30 LPTIM\_LPTIE\_TRIGIE\_Getable

函数名	LPTIM_LPTIE_TRIGIE_Getable
函数原型	FunState LPTIM_LPTIE_TRIGIE_Getable(void)
功能描述	获取外部触发中断状态
输入参数 1	无
输出参数	无
返回值	外部触发中断： DISABLE：禁止外部触发中断 ENABLE：使能外部触发中断

### 11.2.31 LPTIM\_LPTIF\_OVIF\_Clr

函数名	LPTIM_LPTIF_OVIF_Clr
函数原型	void LPTIM_LPTIF_OVIF_Clr(void)
功能描述	清除计数器溢出标志
输入参数 1	无
输出参数	无
返回值	无

### 11.2.32 LPTIM\_LPTIF\_OVIF\_Chk

函数名	LPTIM_LPTIF_OVIF_Chk
函数原型	FlagStatus LPTIM_LPTIF_OVIF_Chk(void)
功能描述	获取计数器溢出标志状态
输入参数 1	无
输出参数	无
返回值	LPTIM 计数器溢出标志： RESET：未产生 LPTIM 计数器溢出 SET：产生 LPTIM 计数器溢出



**11.2.33 LPTIM\_LPTIF\_COMPIF\_Clr**

函数名	LPTIM_LPTIF_COMPIF_Clr
函数原型	void LPTIM_LPTIF_COMPIF_Clr(void)
功能描述	清除比较值匹配标志
输入参数 1	无
输出参数	无
返回值	无

**11.2.34 LPTIM\_LPTIF\_COMPIF\_Chk**

函数名	LPTIM_LPTIF_COMPIF_Chk
函数原型	FlagStatus LPTIM_LPTIF_COMPIF_Chk(void)
功能描述	获取比较值匹配标志状态
输入参数 1	无
输出参数	无
返回值	LPTIM 比较值匹配标志： RESET：未产生 LPTIM 比较值匹配 SET：产生 LPTIM 比较值匹配

**11.2.35 LPTIM\_LPTIF\_TRIGIF\_Clr**

函数名	LPTIM_LPTIF_TRIGIF_Clr
函数原型	void LPTIM_LPTIF_TRIGIF_Clr(void)
功能描述	清除外部触发标志
输入参数 1	无
输出参数	无
返回值	无

**11.2.36 LPTIM\_LPTIF\_TRIGIF\_Chk**

函数名	LPTIM_LPTIF_TRIGIF_Chk
函数原型	FlagStatus LPTIM_LPTIF_TRIGIF_Chk(void)
功能描述	获取外部触发标志状态
输入参数 1	无
输出参数	无
返回值	LPTIM 外部触发标志： RESET：未产生 LPTIM 外部触发



SET: 产生 LPTIM 外部触发

### 11.2.37 LPTIM\_LPTCFG\_FLTEN\_Setable

函数名	LPTIM_LPTCFG_FLTEN_Setable
函数原型	void LPTIM_LPTCFG_FLTEN_Setable(FunState NewState)
功能描述	设置外部 Trigger 滤波使能
输入参数 1	外部 Trigger 滤波使能: DISABLE: 禁止外部 Trigger 滤波 ENABLE: 使能外部 Trigger 滤波
输出参数	无
返回值	无

### 11.2.38 LPTIM\_LPTCFG\_FLTEN\_Getable

函数名	LPTIM_LPTCFG_FLTEN_Getable
函数原型	FunState LPTIM_LPTCFG_FLTEN_Getable(void)
功能描述	获取外部 Trigger 滤波使能状态
输入参数 1	无
输出参数	无
返回值	外部 Trigger 滤波使能: DISABLE: 禁止外部 Trigger 滤波 ENABLE: 使能外部 Trigger 滤波

### 11.2.39 LPTIM\_LPTCFG\_PWM\_Set

函数名	LPTIM_LPTCFG_PWM_Set
函数原型	void LPTIM_LPTCFG_PWM_Set(uint32_t SetValue)
功能描述	设置 PWM 模式选择
输入参数 1	脉宽调制模式 0: 周期方波输出模式 1: PWM 输出模式
输出参数	无
返回值	无

### 11.2.40 LPTIM\_LPTCFG\_PWM\_Get

函数名	LPTIM_LPTCFG_PWM_Get
-----	----------------------



函数原型	uint32_t LPTIM_LPTCFG_PWM_Get(void)
功能描述	获取 PWM 模式选择
输入参数 1	无
输出参数	无
返回值	脉宽调制模式 0: 周期方波输出模式 1: PWM 输出模式

# 12 ETIM

## 12.1 ETIM 寄存器结构

寄存器	描述
ET1CR	ETIMx 控制寄存器
ET2CR	
ET3CR	
ET4CR	
ET1INSEL	ETIMx 输入选择寄存器
ET2INSEL	
ET3INSEL	
ET4INSEL	
ET1PESCALE1	ETIMx 预分频寄存器 1
ET2PESCALE1	
ET3PESCALE1	
ET4PESCALE1	
ET1PESCALE2	ETIMx 预分频寄存器 2
ET2PESCALE2	
ET3PESCALE2	
ET4PESCALE2	
ET1IVR	ETIMx 初值寄存器
ET2IVR	
ET3IVR	
ET4IVR	
ET1CMP	ETIMx 比较寄存器
ET2CMP	
ET3CMP	
ET4CMP	
ET1IE	ETIMx 中断使能寄存器
ET2 IE	
ET3 IE	
ET4 IE	
ET1IF	ETIMx 中断标志寄存器
ET2 IF	
ET3 IF	

ET4 IF

## 12.2 ETIM 库函数

序号	函数名	描述
1	ETIMx_Init	ET 初始化设置
2	ETIMx_Deinit	ET 初始化为缺省值
3	ETIMx_ETxCR_CEN_Setable	设置 ET 计数器启动
4	ETIMx_ETxCR_CEN_Getable	获取 ET 计数器启动状态
5	ETIMx_ETxCR_MOD_Set	设置 ET 计数器工作模式
6	ETIMx_ETxCR_MOD_Get	获取 ET 计数器工作模式
7	ETIMx_ETxCR_EDGESEL_Set	设置 ET 计数器计数模式采沿方式
8	ETIMx_ETxCR_EDGESEL_Get	获取 ET 计数器计数模式采沿方式
9	ETIMx_ETxPESCALE1_Write	设置 ET Signal1(计数源)的预分频
10	ETIMx_ETxPESCALE1_Read	获取 ET Signal1(计数源) 的预分频
11	ETIMx_ETxPESCALE2_Write	设置 ET Signal2(计数源)的预分频
12	ETIMx_ETxPESCALE2_Read	获取 ET Signal2(计数源) 的预分频
13	ETIMx_ETxIVR_Write	设置 ET 计数初值
14	ETIMx_ETxIVR_Read	获取 ET 计数初值
15	ETIMx_ETxCMP_Write	设置 ET 比较值
16	ETIMx_ETxCMP_Read	获取 ET 比较值
17	ETIMx_ETxCR_PWM_Setable	ET 计数器 PWM 输出使能设置
18	ETIMx_ETxCR_PWM_Getable	获取 ET 计数器 PWM 输出使能设置
19	ETIMx_ETxINSEL_SIG1SEL_Set	设置 ET 内部信号 1 源选择
20	ETIMx_ETxINSEL_SIG1SEL_Get	获取 ET 内部信号 1 源选择
21	ETIMx_ETxINSEL_SIG2SEL_Set	设置 ET 内部信号 2 源选择
22	ETIMx_ETxINSEL_SIG2SEL_Get	获取 ET 内部信号 2 源选择
23	ETIMx_ETxCR_CAPEDGE_Set	设置 ET 计数器捕捉沿选择
24	ETIMx_ETxCR_CAPEDGE_Get	获取 ET 计数器捕捉沿选择设置
25	ETIMx_ETxCR_CAPMOD_Set	设置 ET 计数器捕捉模式
26	ETIMx_ETxCR_CAPMOD_Get	获取 ET 计数器捕捉模式
27	ETIMx_ETxCR_CAPONCE_Setable	设置 ET 计数器单次捕捉有效
28	ETIMx_ETxCR_CAPONCE_Getable	获取 ET 计数器单次捕捉有效设置
29	ETIMx_ETxCR_CAPCLR_Setable	设置 ET 计数器带清零捕捉模式
30	ETIMx_ETxCR_CAPCLR_Getable	获取置 ET 计数器带清零捕捉模式设置

31	ETIMx_ETxINSEL_GRP1SEL_Set	设置 ET GROUP1 信号选择
32	ETIMx_ETxINSEL_GRP1SEL_Get	获取 ET GROUP1 信号选择
33	ETIMx_ETxINSEL_GRP2SEL_Set	设置 ET GROUP2 信号选择
34	ETIMx_ETxINSEL_GRP2SEL_Get	获取 ET GROUP2 信号选择
35	ETIMx_ETxCR_CASEN_Setable	设置 ET 计数器级联功能
36	ETIMx_ETxCR_CASEN_Getable	获取 ET 计数器级联功能设置
37	ETIMx_ETxIE_CMPIE_Setable	设置 ET 比较中断
38	ETIMx_ETxIE_CMPIE_Getable	获取 ET 比较中断设置
39	ETIMx_ETxIE_CAPIE_Setable	设置 ET 捕捉中断
40	ETIMx_ETxIE_CAPIE_Getable	获取 ET 捕捉中断
41	ETIMx_ETxIE_OVIE_Setable	设置 ET 溢出中断
42	ETIMx_ETxIE_OVIE_Getable	获取 ET 溢出中断
43	ETIMx_ETxIF_CMPIF_Clr	清除 ET 比较中断标志
44	ETIMx_ETxIF_CMPIF_Chk	获取 ET 比较中断标志
45	ETIMx_ETxIF_CAPIF_Clr	清除 ET 产生捕捉信号标志
46	ETIMx_ETxIF_CAPIF_Chk	获取 ET 产生捕捉信号标志
47	ETIMx_ETxIF_OVIF_Clr	清除 ET 产生溢出信号标志
48	ETIMx_ETxIF_OVIF_Chk	获取 ET 产生溢出信号标志
49	ETIMx_ETxIF_EDGESTA_Chk	获取 ET 捕捉沿状态
50	ETIMx_ETxCR_EXFLT_Setable	ET 计数器外部输入引脚数字滤波功能设置
51	ETIMx_ETxCR_EXFLT_Getable	获取 ET 计数器外部输入引脚数字滤波功能设置

### 12.2.1 ETIMx\_Init

函数名	ETIMx_Init
函数原型	void ETIMx_Init(ETIMx_Type* ETIMx, ETIM_InitTypeDef* para)
功能描述	ET 初始化配置
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	<div>ETIM 配置参数:</div> <div><div>信号来源参数:</div><div><div>内部信号 1 源选择:</div><div>0: 扩展定时器 3 的内部信号 1 选择 GROUP2</div><div>1: 扩展定时器 3 的内部信号 1 选择 GROUP1</div></div><div><div>内部信号 2 源选择:</div><div>0: 扩展定时器 3 的内部信号 2 选择 GROUP2</div><div>1: 扩展定时器 3 的内部信号 2 选择 GROUP1</div></div></div>



	<p>GROUP1 信号选择控制:</p> <p>ET1</p> <p>00: APBCLK</p> <p>01: XTLF</p> <p>10: RCLP</p> <p>11: ET1_IN0</p> <p>ET2</p> <p>00: APBCLK</p> <p>01: XTLF</p> <p>10: RCLP</p> <p>11: ET2_IN0</p> <p>ET3</p> <p>00: APBCLK</p> <p>01: ET3_IN0</p> <p>10: RTCSEC</p> <p>11: RCLP</p> <p>ET4</p> <p>00: APBCLK</p> <p>01: ET4_IN0</p> <p>10: RTC64HZ</p> <p>11: LPTO</p>	
	<p>GROUP2 信号选择控制:</p> <p>ET1</p> <p>000: UART0_RX</p> <p>001: UART1_RX</p> <p>010: XTLF</p> <p>011: ET1_IN1</p> <p>100: ET1_IN2</p> <p>101: CMP1O</p> <p>110: CMP2O</p> <p>111: LPTO</p> <p>ET2</p> <p>000: UART2_RX</p> <p>001: UART3_RX</p> <p>010: XTLF</p> <p>011: ET2_IN1</p> <p>100: ET2_IN2</p> <p>101: CMP1O</p> <p>110: CMP2O</p> <p>111: LPTO</p> <p>ET3</p> <p>000: ET3_IN1</p> <p>001: XTLF</p> <p>010: UART4_RX</p>	



	011: UART5_RX 100: RTCSEC 101~111: RFU ET4 000: ET4_IN1 001: XTLF 010: UART_RX2 011: UART_RX0 100: CMP1O 101: CMP2O 110: RTCSEC 111: LPTO	
	ETx 预分频寄存器 1: 00~FF, 00 表示不分频, FF 表示 256 分频	
	ETx 预分频寄存器 2: 00~FF, 00 表示不分频, FF 表示 256 分频	
	控制类参数:	
	引脚输入数字滤波使能: DISABLE: 禁止引脚输入数字滤波 ENABLE: 使能引脚输入数字滤波	
	工作模式选择: 0: 定时/计数模式 1: 捕捉模式	
	扩展定时器级联使能: DISABLE: 禁止扩展定时器级联 ENABLE: 使能扩展定时器级联	
	计数模式采沿方式选择: 0: 上升沿采样 1: 下降沿采样	
	PWM 输出控制: DISABLE: 禁止 PWM 输出 ENABLE: 使能 PWM 输出	
	捕捉模式控制: 0: 脉冲周期捕捉 1: 脉宽捕捉	
	带清零捕捉模式控制: 0: 捕捉不清零 1: 时间触发捕捉, 使能后计数器保持 0, 捕捉到第一个有消沿之后 timer 才开始计数	
	单次捕捉控制: 0: 连续捕捉 1: 单次捕捉有效, 在捕捉到一次脉冲周期后计数器停止, 若再需要捕捉需重新启动	
	捕捉沿选择: 0: 上升沿捕捉	



	1: 下降沿捕捉	
	ETx 初值寄存器: ETx 初值	
	ETx 比较寄存器: ETx 比较值	
	扩展定时器比较中断使能:	
	DISABLE: 禁止扩展定时器比较中断	
	ENABLE: 使能扩展定时器比较中断	
	扩展定时器捕捉中断使能:	
	DISABLE: 禁止扩展定时器捕捉中断	
ENABLE: 使能扩展定时器捕捉中断		
	扩展定时器溢出中断使能:	
	DISABLE: 禁止扩展定时器溢出中断	
	ENABLE: 使能扩展定时器溢出中断	
输出参数	无	
返回值	无	

### 12.2.2 ETIMx\_Deinit

函数名	ETIMx_Deinit
函数原型	void ETIMx_Deinit(ETIMx_Type* ETIMx)
功能描述	ET 初始化为默认值
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	无

### 12.2.3 ETIMx\_ETxCR\_CEN\_Setable

函数名	ETIMx_ETxCR_CEN_Setable
函数原型	void ETIMx_ETxCR_CEN_Setable(ETIMx_Type* ETIMx, FunState NewState)
功能描述	配置 ET 计数器启动
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	DISABLE: 停止定时器 ENABLE: 启动定时器
输出参数	无
返回值	无

### 12.2.4 ETIMx\_ETxCR\_CEN\_Getable

函数名	ETIMx_ETxCR_CEN_Getable
函数原型	FunState ETIMx_ETxCR_CEN_Getable(ETIMx_Type* ETIMx)
功能描述	获取 ET 计数器启动状态

输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	DISABLE: 停止定时器 ENABLE: 启动定时器

### 12.2.5 ETIMx\_ETxCR\_MOD\_Set

函数名	ETIMx_ETxCR_MOD_Set
函数原型	void ETIMx_ETxCR_MOD_Set(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	配置 ET 计数器工作模式
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 计数器工作模式: ETIMx_ETxCR_MOD_COUNTER: 计数模式 ETIMx_ETxCR_MOD_CAPTURE: 捕捉模式
输出参数	无
返回值	无

### 12.2.6 ETIMx\_ETxCR\_MOD\_Get

函数名	ETIMx_ETxCR_MOD_Get
函数原型	uint32_t ETIMx_ETxCR_MOD_Get(ETIMx_Type* ETIMx)
功能描述	获取 ET 计数器工作模式
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 计数器工作模式: ETIMx_ETxCR_MOD_COUNTER: 计数模式 ETIMx_ETxCR_MOD_CAPTURE: 捕捉模式

### 12.2.7 ETIMx\_ETxCR\_EDGESEL\_Set

函数名	ETIMx_ETxCR_EDGESEL_Set
函数原型	void ETIMx_ETxCR_EDGESEL_Set(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	配置 ET 计数器计数模式采沿方式
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 计数器计数模式采沿方式: ETIMx_ETxCR_EDGESEL_RISING: 计数模式采上升沿 ETIMx_ETxCR_EDGESEL_FALLING: 计数模式采下降沿
输出参数	无

返回值	无
-----	---

### 12.2.8 ETIMx\_ETxCR\_EDGESEL\_Get

函数名	ETIMx_ETxCR_EDGESEL_Get
函数原型	uint32_t ETIMx_ETxCR_EDGESEL_Get(ETIMx_Type* ETIMx)
功能描述	获取 ET 计数器计数模式采沿方式
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 计数器计数模式采沿方式: ETIMx_ETxCR_EDGESEL_RISING: 计数模式采上升沿 ETIMx_ETxCR_EDGESEL_FALLING: 计数模式采下降沿

### 12.2.9 ETIMx\_ETxPESCALE1\_Write

函数名	ETIMx_ETxPESCALE1_Write
函数原型	void ETIMx_ETxPESCALE1_Write(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	配置 ET Signal1(计数源)的预分频
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET Signal1(计数源)的预分频: 00-FF(00 表示 1 分频, FF 表示 256 分频)
输出参数	无
返回值	无

### 12.2.10 ETIMx\_ETxPESCALE1\_Read

函数名	ETIMx_ETxPESCALE1_Read
函数原型	uint32_t ETIMx_ETxPESCALE1_Read(ETIMx_Type* ETIMx)
功能描述	获取 ET Signal1(计数源) 的预分频
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET Signal1(计数源)的预分频: 00-FF(00 表示 1 分频, FF 表示 256 分频)

### 12.2.11 ETIMx\_ETxPESCALE2\_Write

函数名	ETIMx_ETxPESCALE2_Write
函数原型	void ETIMx_ETxPESCALE2_Write(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	配置 ET Signal2(计数源)的预分频
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4

输入参数 2	ET Signal2(计数源)的预分频: 00-FF(00 表示 1 分频, FF 表示 256 分频)
输出参数	无
返回值	无

### 12.2.12 ETIMx\_ETxPESCALE2\_Read

函数名	ETIMx_ETxPESCALE2_Read
函数原型	uint32_t ETIMx_ETxPESCALE2_Read(ETIMx_Type* ETIMx)
功能描述	获取 ET Signal2(计数源) 的预分频
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET Signal2(计数源)的预分频: 00-FF(00 表示 1 分频, FF 表示 256 分频)

### 12.2.13 ETIMx\_ETxIVR\_Write

函数名	ETIMx_ETxIVR_Write
函数原型	void ETIMx_ETxIVR_Write(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	配置 ET 计数初值
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数 2	ET 计数初值: 00-FFFF
输出参数	无
返回值	无

### 12.2.14 ETIMx\_ETxIVR\_Read

函数名	ETIMx_ETxIVR_Read
函数原型	uint32_t ETIMx_ETxIVR_Read(ETIMx_Type* ETIMx)
功能描述	获取 ET 计数初值
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 计数初值: 00-FFFF

### 12.2.15 ETIMx\_ETxCMP\_Write

函数名	ETIMx_ETxCMP_Write
函数原型	void ETIMx_ETxCMP_Write(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	配置 ET 比较值
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4

输入参数 2	ET 比较值: 00-FFFF
输出参数	无
返回值	无

### 12.2.16 ETIMx\_ETxCMP\_Read

函数名	ETIMx_ETxCMP_Read
函数原型	uint32_t ETIMx_ETxCMP_Read(ETIMx_Type* ETIMx)
功能描述	获取 ET 比较值
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 比较值: 00-FFFF

### 12.2.17 ETIMx\_ETxCR\_PWM\_Setable

函数名	ETIMx_ETxCR_PWM_Setable
函数原型	void ETIMx_ETxCR_PWM_Setable(ETIMx_Type* ETIMx, FunState NewState)
功能描述	ET 计数器 PWM 输出使能配置
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 计数器 PWM 输出使能: DISABLE: 禁止 PWM 输出 ENABLE: 使能 PWM 输出
输出参数	无
返回值	无

### 12.2.18 ETIMx\_ETxCR\_PWM\_Getable

函数名	ETIMx_ETxCR_PWM_Getable
函数原型	FunState ETIMx_ETxCR_PWM_Getable(ETIMx_Type* ETIMx)
功能描述	获取 ET 计数器 PWM 输出使能配置
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 计数器 PWM 输出使能: DISABLE: 禁止 PWM 输出 ENABLE: 使能 PWM 输出

**12.2.19 ETIMx\_ETxINSEL\_SIG1SEL\_Set**

函数名	ETIMx_ETxINSEL_SIG1SEL_Set
函数原型	void ETIMx_ETxINSEL_SIG1SEL_Set(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	配置 ET 内部信号 1 源选择
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 内部信号 1 源选择: ETIMx_ETxINSEL_SIG1SEL_GROUP1: 扩展定时器的内部信号 1 选择 Group1 ETIMx_ETxINSEL_SIG1SEL_GROUP2: 扩展定时器的内部信号 1 选择 Group2
输出参数	无
返回值	无

**12.2.20 ETIMx\_ETxINSEL\_SIG1SEL\_Get**

函数名	ETIMx_ETxINSEL_SIG1SEL_Get
函数原型	uint32_t ETIMx_ETxINSEL_SIG1SEL_Get(ETIMx_Type* ETIMx)
功能描述	获取 ET 内部信号 1 源选择
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 内部信号 1 源选择: ETIMx_ETxINSEL_SIG1SEL_GROUP1: 扩展定时器的内部信号 1 选择 Group1 ETIMx_ETxINSEL_SIG1SEL_GROUP2: 扩展定时器的内部信号 1 选择 Group2

**12.2.21 ETIMx\_ETxINSEL\_SIG2SEL\_Set**

函数名	ETIMx_ETxINSEL_SIG2SEL_Set
函数原型	void ETIMx_ETxINSEL_SIG2SEL_Set(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	配置 ET 内部信号 2 源选择
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 内部信号 2 源选择: ETIMx_ETxINSEL_SIG2SEL_GROUP2: 扩展定时器 3 的内部信号 2 选择 Group2 ETIMx_ETxINSEL_SIG2SEL_GROUP1: 扩展定时器 3 的内部信号 2 选择 Group1
输出参数	无
返回值	无

**12.2.22 ETIMx\_ETxINSEL\_SIG2SEL\_Get**

函数名	ETIMx_ETxINSEL_SIG2SEL_Get
-----	----------------------------

函数原型	uint32_t ETIMx_ETxINSEL_SIG2SEL_Get(ETIMx_Type* ETIMx)
功能描述	获取 ET 内部信号 2 源选择
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 内部信号 2 源选择: ETIMx_ETxINSEL_SIG2SEL_GROUP2: 扩展定时器 3 的内部信号 2 选择 Group2 ETIMx_ETxINSEL_SIG2SEL_GROUP1: 扩展定时器 3 的内部信号 2 选择 Group1

### 12.2.23 ETIMx\_ETxCR\_CAPEdge\_Set

函数名	ETIMx_ETxCR_CAPEdge_Set
函数原型	void ETIMx_ETxCR_CAPEdge_Set(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	配置 ET 计数器捕捉沿选择
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 计数器捕捉沿选择: ETIMx_ETxCR_CAPEdge_RISING: 周期捕捉模式时上沿捕捉 ETIMx_ETxCR_CAPEdge_FALLING: 周期捕捉模式时下沿捕捉
输出参数	无
返回值	无

### 12.2.24 ETIMx\_ETxCR\_CAPEdge\_Get

函数名	ETIMx_ETxCR_CAPEdge_Get
函数原型	uint32_t ETIMx_ETxCR_CAPEdge_Get(ETIMx_Type* ETIMx)
功能描述	获取 ET 计数器捕捉沿选择配置
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 计数器捕捉沿选择: ETIMx_ETxCR_CAPEdge_RISING: 周期捕捉模式时上沿捕捉 ETIMx_ETxCR_CAPEdge_FALLING: 周期捕捉模式时下沿捕捉

### 12.2.25 ETIMx\_ETxCR\_CAPMOD\_Set

函数名	ETIMx_ETxCR_CAPMOD_Set
函数原型	void ETIMx_ETxCR_CAPMOD_Set(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	配置 ET 计数器捕捉模式
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 计数器捕捉模式:



	ETIMx_ETxCR_CAPMOD_PERIOD: 脉冲周期捕捉 ETIMx_ETxCR_CAPMOD_PULSE: 脉宽捕捉
输出参数	无
返回值	无

### 12.2.26 ETIMx\_ETxCR\_CAPMOD\_Get

函数名	ETIMx_ETxCR_CAPMOD_Get
函数原型	uint32_t ETIMx_ETxCR_CAPMOD_Get(ETIMx_Type* ETIMx)
功能描述	获取 ET 计数器捕捉模式
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 计数器捕捉模式: ETIMx_ETxCR_CAPMOD_PERIOD: 脉冲周期捕捉 ETIMx_ETxCR_CAPMOD_PULSE: 脉宽捕捉

### 12.2.27 ETIMx\_ETxCR\_CAPONCE\_Setable

函数名	ETIMx_ETxCR_CAPONCE_Setable
函数原型	void ETIMx_ETxCR_CAPONCE_Setable(ETIMx_Type* ETIMx, FunState NewState)
功能描述	配置 ET 计数器单次捕捉有效
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 计数器单次捕捉有效: DISABLE: 连续捕捉 ENABLE: 单次捕捉有效
输出参数	无
返回值	无

### 12.2.28 ETIMx\_ETxCR\_CAPONCE\_Getable

函数名	ETIMx_ETxCR_CAPONCE_Getable
函数原型	FunState ETIMx_ETxCR_CAPONCE_Getable(ETIMx_Type* ETIMx)
功能描述	获取 ET 计数器单次捕捉有效配置
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 计数器单次捕捉有效: DISABLE: 连续捕捉 ENABLE: 单次捕捉有效



### 12.2.29 ETIMx\_ETxCR\_CAPCLR\_Setable

函数名	ETIMx_ETxCR_CAPCLR_Setable
函数原型	void ETIMx_ETxCR_CAPCLR_Setable(ETIMx_Type* ETIMx, FunState NewState)
功能描述	配置 ET 计数器带清零捕捉模式
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 计数器带清零捕捉模式: DISABLE: 捕捉不清零, 计数器一直自由计数 ENABLE: 事件触发捕捉, 使能后计数器保持 0, 捕捉到第一个有效沿之后 timer 才开始计数
输出参数	无
返回值	无

### 12.2.30 ETIMx\_ETxCR\_CAPCLR\_Getable

函数名	ETIMx_ETxCR_CAPCLR_Getable
函数原型	FunState ETIMx_ETxCR_CAPCLR_Getable(ETIMx_Type* ETIMx)
功能描述	获取 ET 计数器带清零捕捉模式配置
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 计数器带清零捕捉模式: DISABLE: 捕捉不清零, 计数器一直自由计数 ENABLE: 事件触发捕捉, 使能后计数器保持 0, 捕捉到第一个有效沿之后 timer 才开始计数

### 12.2.31 ETIMx\_ETxINSEL\_GRP1SEL\_Set

函数名	ETIMx_ETxINSEL_GRP1SEL_Set
函数原型	void ETIMx_ETxINSEL_GRP1SEL_Set(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	设置 ET GROUP1 信号选择
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET GROUP1 信号选择: ETIMx_ETxINSEL_GRP1SEL_ET1_APBCLK: APBCLK ETIMx_ETxINSEL_GRP1SEL_ET1_XTLF: XTLF ETIMx_ETxINSEL_GRP1SEL_ET1_RCLP: RCLP ETIMx_ETxINSEL_GRP1SEL_ET1_ET1_IN0: ET1_IN0 ETIMx_ETxINSEL_GRP1SEL_ET2_APBCLK: APBCLK

	ETIMx_ETxINSEL_GRP1SEL_ET2_XTLF: XTLF ETIMx_ETxINSEL_GRP1SEL_ET2_RCLP: RCLP ETIMx_ETxINSEL_GRP1SEL_ET2_ET2_IN0: ET2_IN0 ETIMx_ETxINSEL_GRP1SEL_ET3_APBCLK: APBCLK ETIMx_ETxINSEL_GRP1SEL_ET3_ET3_IN0: ET3_IN0 ETIMx_ETxINSEL_GRP1SEL_ET3_RTCSEC: RTCSEC ETIMx_ETxINSEL_GRP1SEL_ET3_RCLP: RCLP ETIMx_ETxINSEL_GRP1SEL_ET4_APBCLK: APBCLK ETIMx_ETxINSEL_GRP1SEL_ET4_ET4_IN0: ET4_IN0 ETIMx_ETxINSEL_GRP1SEL_ET4_RTC64HZ: RTC64HZ ETIMx_ETxINSEL_GRP1SEL_ET4_LPTO: LPTO
输出参数	无
返回值	无

### 12.2.32 ETIMx\_ETxINSEL\_GRP1SEL\_Get

函数名	ETIMx_ETxINSEL_GRP1SEL_Get
函数原型	uint32_t ETIMx_ETxINSEL_GRP1SEL_Get(ETIMx_Type* ETIMx)
功能描述	获取 ET GROUP1 信号选择
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET GROUP1 信号选择: ETIMx_ETxINSEL_GRP1SEL_ET1_APBCLK: APBCLK ETIMx_ETxINSEL_GRP1SEL_ET1_XTLF: XTLF ETIMx_ETxINSEL_GRP1SEL_ET1_RCLP: RCLP ETIMx_ETxINSEL_GRP1SEL_ET1_ET1_IN0: ET1_IN0 ETIMx_ETxINSEL_GRP1SEL_ET2_APBCLK: APBCLK ETIMx_ETxINSEL_GRP1SEL_ET2_XTLF: XTLF ETIMx_ETxINSEL_GRP1SEL_ET2_RCLP: RCLP ETIMx_ETxINSEL_GRP1SEL_ET2_ET2_IN0: ET2_IN0 ETIMx_ETxINSEL_GRP1SEL_ET3_APBCLK: APBCLK ETIMx_ETxINSEL_GRP1SEL_ET3_ET3_IN0: ET3_IN0 ETIMx_ETxINSEL_GRP1SEL_ET3_RTCSEC: RTCSEC ETIMx_ETxINSEL_GRP1SEL_ET3_RCLP: RCLP ETIMx_ETxINSEL_GRP1SEL_ET4_APBCLK: APBCLK ETIMx_ETxINSEL_GRP1SEL_ET4_ET4_IN0: ET4_IN0 ETIMx_ETxINSEL_GRP1SEL_ET4_RTC64HZ: RTC64HZ ETIMx_ETxINSEL_GRP1SEL_ET4_LPTO: LPTO

### 12.2.33 ETIMx\_ETxINSEL\_GRP2SEL\_Set

函数名	ETIMx_ETxINSEL_GRP2SEL_Set
函数原型	void ETIMx_ETxINSEL_GRP2SEL_Set(ETIMx_Type* ETIMx, uint32_t SetValue)
功能描述	设置 ET GROUP2 信号选择
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET GROUP2 信号选择: ETIMx_ETxINSEL_GRP2SEL_ET1_UART0_RX: UART0_RX ETIMx_ETxINSEL_GRP2SEL_ET1_UART1_RX: UART1_RX ETIMx_ETxINSEL_GRP2SEL_ET1_XTLF: XTLF ETIMx_ETxINSEL_GRP2SEL_ET1_ET1_IN1: ET1_IN1 ETIMx_ETxINSEL_GRP2SEL_ET1_ET1_IN2: ET1_IN2 ETIMx_ETxINSEL_GRP2SEL_ET1_CMP1O: CMP1O (比较器 1 输出) ETIMx_ETxINSEL_GRP2SEL_ET1_CMP2O: CMP2O (比较器 2 输出) ETIMx_ETxINSEL_GRP2SEL_ET1_LPTO: LPTO ETIMx_ETxINSEL_GRP2SEL_ET2_UART2_RX: UART2_RX ETIMx_ETxINSEL_GRP2SEL_ET2_UART3_RX: UART3_RX ETIMx_ETxINSEL_GRP2SEL_ET2_XTLF: XTLF ETIMx_ETxINSEL_GRP2SEL_ET2_ET2_IN1: ET2_IN1 ETIMx_ETxINSEL_GRP2SEL_ET2_ET2_IN2: ET2_IN2 ETIMx_ETxINSEL_GRP2SEL_ET2_CMP1O: CMP1O (比较器 1 输出) ETIMx_ETxINSEL_GRP2SEL_ET2_CMP2O: CMP2O (比较器 2 输出) ETIMx_ETxINSEL_GRP2SEL_ET2_LPTO: LPTO ETIMx_ETxINSEL_GRP2SEL_ET3_ET3_IN12: ET3_IN1 ETIMx_ETxINSEL_GRP2SEL_ET3_XTLF: XTLF ETIMx_ETxINSEL_GRP2SEL_ET3_UART4_RX: UART4_RX ETIMx_ETxINSEL_GRP2SEL_ET3_UART5_RX: UART5_RX ETIMx_ETxINSEL_GRP2SEL_ET3_RTCSEC: RTCSEC ETIMx_ETxINSEL_GRP2SEL_ET4_ET4_IN1: ET4_IN1 ETIMx_ETxINSEL_GRP2SEL_ET4_XTLF: XTLF ETIMx_ETxINSEL_GRP2SEL_ET4_UART_RX22: UART_RX2 ETIMx_ETxINSEL_GRP2SEL_ET4_UART_RX0: UART_RX0 ETIMx_ETxINSEL_GRP2SEL_ET4_CMP1O: CMP1O (比较器 1 输出) ETIMx_ETxINSEL_GRP2SEL_ET4_CMP2O: CMP2O (比较器 2 输出) ETIMx_ETxINSEL_GRP2SEL_ET4_RTCSEC: RTCSEC ETIMx_ETxINSEL_GRP2SEL_ET4_LPTO: LPTO
输出参数	无

返回值	无
-----	---

### 12.2.34 ETIMx\_ETxINSEL\_GRP2SEL\_Get

函数名	ETIMx_ETxINSEL_GRP2SEL_Get
函数原型	uint32_t ETIMx_ETxINSEL_GRP2SEL_Get(ETIMx_Type* ETIMx)
功能描述	获取 ET GROUP2 信号选择
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET GROUP2 信号选择: ETIMx_ETxINSEL_GRP2SEL_ET1_UART0_RX: UART0_RX ETIMx_ETxINSEL_GRP2SEL_ET1_UART1_RX: UART1_RX ETIMx_ETxINSEL_GRP2SEL_ET1_XTLF: XTLF ETIMx_ETxINSEL_GRP2SEL_ET1_ET1_IN1: ET1_IN1 ETIMx_ETxINSEL_GRP2SEL_ET1_ET1_IN2: ET1_IN2 ETIMx_ETxINSEL_GRP2SEL_ET1_CMP1O: CMP1O (比较器 1 输出) ETIMx_ETxINSEL_GRP2SEL_ET1_CMP2O: CMP2O (比较器 2 输出) ETIMx_ETxINSEL_GRP2SEL_ET1_LPTO: LPTO ETIMx_ETxINSEL_GRP2SEL_ET2_UART2_RX: UART2_RX ETIMx_ETxINSEL_GRP2SEL_ET2_UART3_RX: UART3_RX ETIMx_ETxINSEL_GRP2SEL_ET2_XTLF: XTLF ETIMx_ETxINSEL_GRP2SEL_ET2_ET2_IN1: ET2_IN1 ETIMx_ETxINSEL_GRP2SEL_ET2_ET2_IN2: ET2_IN2 ETIMx_ETxINSEL_GRP2SEL_ET2_CMP1O: CMP1O (比较器 1 输出) ETIMx_ETxINSEL_GRP2SEL_ET2_CMP2O: CMP2O (比较器 2 输出) ETIMx_ETxINSEL_GRP2SEL_ET2_LPTO: LPTO ETIMx_ETxINSEL_GRP2SEL_ET3_ET3_IN12: ET3_IN1 ETIMx_ETxINSEL_GRP2SEL_ET3_XTLF: XTLF ETIMx_ETxINSEL_GRP2SEL_ET3_UART4_RX: UART4_RX ETIMx_ETxINSEL_GRP2SEL_ET3_UART5_RX: UART5_RX ETIMx_ETxINSEL_GRP2SEL_ET3_RTCSEC: RTCSEC ETIMx_ETxINSEL_GRP2SEL_ET4_ET4_IN1: ET4_IN1 ETIMx_ETxINSEL_GRP2SEL_ET4_XTLF: XTLF ETIMx_ETxINSEL_GRP2SEL_ET4_UART_RX22: UART_RX2 ETIMx_ETxINSEL_GRP2SEL_ET4_UART_RX0: UART_RX0 ETIMx_ETxINSEL_GRP2SEL_ET4_CMP1O: CMP1O (比较器 1 输出) ETIMx_ETxINSEL_GRP2SEL_ET4_CMP2O: CMP2O (比较器 2 输出) ETIMx_ETxINSEL_GRP2SEL_ET4_RTCSEC: RTCSEC

ETIMx\_ETxINSEL\_GRP2SEL\_ET4\_LPTO: LPTO

**12.2.35 ETIMx\_ETxCR\_CASEN\_Setable**

函数名	ETIMx_ETxCR_CASEN_Setable
函数原型	void ETIMx_ETxCR_CASEN_Setable(ETIMx_Type* ETIMx, FunState NewState)
功能描述	配置 ET 计数器级联功能
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 计数器级联功能使能: DISABLE: 16bit 定时器独立工作 ENABLE: ET1 (ET3) 和 ET2 (ET4) 级联成 32bit 定时器
输出参数	无
返回值	无

**12.2.36 ETIMx\_ETxCR\_CASEN\_Getable**

函数名	ETIMx_ETxCR_CASEN_Getable
函数原型	FunState ETIMx_ETxCR_CASEN_Getable(ETIMx_Type* ETIMx)
功能描述	获取 ET 计数器级联功能配置
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 计数器级联功能使能: DISABLE: 16bit 定时器独立工作 ENABLE: ET1 (ET3) 和 ET2 (ET4) 级联成 32bit 定时器

**12.2.37 ETIMx\_ETxIE\_CMPIE\_Setable**

函数名	ETIMx_ETxIE_CMPIE_Setable
函数原型	void ETIMx_ETxIE_CMPIE_Setable(ETIMx_Type* ETIMx, FunState NewState)
功能描述	配置 ET 比较中断
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 比较中断: DISABLE: 禁止比较中断 ENABLE: 使能比较中断
输出参数	无
返回值	无

**12.2.38 ETIMx\_ETxIE\_CMPIE\_Getable**

函数名	ETIMx_ETxIE_CMPIE_Getable
函数原型	FunState ETIMx_ETxIE_CMPIE_Getable(ETIMx_Type* ETIMx)
功能描述	获取 ET 比较中断配置
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 比较中断: DISABLE: 禁止比较中断 ENABLE: 使能比较中断

**12.2.39 ETIMx\_ETxIE\_CAPIE\_Setable**

函数名	ETIMx_ETxIE_CAPIE_Setable
函数原型	void ETIMx_ETxIE_CAPIE_Setable(ETIMx_Type* ETIMx, FunState NewState)
功能描述	配置 ET 捕捉中断
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 捕捉中断: DISABLE: 禁止捕捉中断 ENABLE: 使能捕捉中断
输出参数	无
返回值	无

**12.2.40 ETIMx\_ETxIE\_CAPIE\_Getable**

函数名	ETIMx_ETxIE_CAPIE_Getable
函数原型	FunState ETIMx_ETxIE_CAPIE_Getable(ETIMx_Type* ETIMx)
功能描述	获取 ET 捕捉中断
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 捕捉中断: DISABLE: 禁止捕捉中断 ENABLE: 使能捕捉中断

**12.2.41 ETIMx\_ETxIE\_OVIE\_Setable**

函数名	ETIMx_ETxIE_OVIE_Setable
函数原型	void ETIMx_ETxIE_OVIE_Setable(ETIMx_Type* ETIMx, FunState NewState)

功能描述	配置 ET 溢出中断
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 溢出中断: DISABLE: 禁止溢出中断 ENABLE: 使能溢出中断
输出参数	无
返回值	无

#### 12.2.42 ETIMx\_ETxIE\_OVIE\_Getable

函数名	ETIMx_ETxIE_OVIE_Getable
函数原型	FunState ETIMx_ETxIE_OVIE_Getable(ETIMx_Type* ETIMx)
功能描述	获取 ET 溢出中断
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 溢出中断: DISABLE: 禁止溢出中断 ENABLE: 使能溢出中断

#### 12.2.43 ETIMx\_ETxIF\_CMPIF\_Clr

函数名	ETIMx_ETxIF_CMPIF_Clr
函数原型	void ETIMx_ETxIF_CMPIF_Clr(ETIMx_Type* ETIMx)
功能描述	清除 ET 比较中断标志
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	无

#### 12.2.44 ETIMx\_ETxIF\_CMPIF\_Chk

函数名	ETIMx_ETxIF_CMPIF_Chk
函数原型	FlagStatus ETIMx_ETxIF_CMPIF_Chk(ETIMx_Type* ETIMx)
功能描述	获取 ET 比较中断标志
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 比较中断标志: RESET: 未发生比较中断 SET: 发生比较中断

### 12.2.45 ETIMx\_ETxIF\_CAPIF\_Clr

函数名	ETIMx_ETxIF_CAPIF_Clr
函数原型	void ETIMx_ETxIF_CAPIF_Clr(ETIMx_Type* ETIMx)
功能描述	清除 ET 产生捕捉信号标志
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	无

### 12.2.46 ETIMx\_ETxIF\_CAPIF\_Chk

函数名	ETIMx_ETxIF_CAPIF_Chk
函数原型	FlagStatus ETIMx_ETxIF_CAPIF_Chk(ETIMx_Type* ETIMx)
功能描述	获取 ET 产生捕捉信号标志
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 产生捕捉信号标志: RESET: 未捕捉到指定的沿 SET: 捕捉到指定的沿

### 12.2.47 ETIMx\_ETxIF\_OVIF\_Clr

函数名	ETIMx_ETxIF_OVIF_Clr
函数原型	void ETIMx_ETxIF_OVIF_Clr(ETIMx_Type* ETIMx)
功能描述	清除 ET 产生溢出信号标志
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	无

### 12.2.48 ETIMx\_ETxIF\_OVIF\_Chk

函数名	ETIMx_ETxIF_OVIF_Chk
函数原型	FlagStatus ETIMx_ETxIF_OVIF_Chk(ETIMx_Type* ETIMx)
功能描述	获取 ET 产生溢出信号标志
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 产生溢出信号标志:



	RESET: 未产生溢出 SET: 产生计数溢出
--	-----------------------------

#### 12.2.49 ETIMx\_ETxIF\_EDGESTA\_Chk

函数名	ETIMx_ETxIF_EDGESTA_Chk
函数原型	FlagStatus ETIMx_ETxIF_EDGESTA_Chk(ETIMx_Type* ETIMx)
功能描述	获取 ET 捕捉沿状态
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 捕捉沿状态: RESET: 脉冲宽度捕捉模式时表示捕捉到上沿 SET: 脉冲宽度捕捉模式时表示捕捉到下沿

#### 12.2.50 ETIMx\_ETxCR\_EXFLT\_Setable

函数名	ETIMx_ETxCR_EXFLT_Setable
函数原型	void ETIMx_ETxCR_EXFLT_Setable(ETIMx_Type* ETIMx, FunState NewState)
功能描述	ET 计数器外部输入引脚数字滤波功能配置
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输入参数 2	ET 计数器外部输入引脚数字滤波功能使能: DISABLE: 禁止引脚输入信号数字滤波 ENABLE: 使能引脚输入信号数字滤波
输出参数	无
返回值	无

#### 12.2.51 ETIMx\_ETxCR\_EXFLT\_Getable

函数名	ETIMx_ETxCR_EXFLT_Getable
函数原型	FunState ETIMx_ETxCR_EXFLT_Getable(ETIMx_Type* ETIMx)
功能描述	获取 ET 计数器外部输入引脚数字滤波功能配置
输入参数 1	定时器名称: ETIM1/ETIM2/ETIM3/ETIM4
输出参数	无
返回值	ET 计数器外部输入引脚数字滤波功能使能: DISABLE: 禁止引脚输入信号数字滤波 ENABLE: 使能引脚输入信号数字滤波

# 13 RTC

## 13.1 RTC 结构

寄存器	描述
RTCWE	RTC 写使能寄存器
RTCIE	RTC 中断使能
RTCIF	RTC 中断标志
BCDSEC	RTC BCD 时间秒寄存器
BCDMIN	RTC BCD 时间分寄存器
BCDHOURL	RTC BCD 时间小时寄存器
BCDDATE	RTC BCD 时间日寄存器
BCDWEEK	RTC BCD 时间周寄存器
BCDMONTH	RTC BCD 时间月寄存器
BCDYEAR	RTC BCD 时间年寄存器
ALARM	RTC 闹钟寄存器
FSEL	RTC 时钟信号输出控制寄存器
ADJUST	RTC 数值调整寄存器
ADSIGN	RTC 数值调整方向寄存器
PR1SEN	RTC 虚拟调校使能寄存器
MSECCNT	RTC 毫秒计数寄存器
STAMPEN	RTC 时间戳使能寄存器
CLKSTAMP0R	RTC 上升沿时间戳 0 寄存器
CALSTAMP0R	RTC 上升沿日历戳 0 寄存器
CLKSTAMP0F	RTC 下降沿时间戳 0 寄存器
CALSTAMP0F	RTC 下降沿日历戳 0 寄存器
CLKSTAMP1R	RTC 上升沿时间戳 1 寄存器
CALSTAMP1R	RTC 上升沿日历戳 1 寄存器
CLKSTAMP1F	RTC 下降沿时间戳 1 寄存器
CALSTAMP1F	RTC 下降沿日历戳 1 寄存器

## 13.2 RTC 库函数

序号	函数名	描述
1	RTC_Deinit	RTC 初始化为缺省值



2	RTC_RTCWE_Write	写入 0xACACACAC, 允许 CPU 向 RTC 的 BCD 时间寄存器写入初值
3	RTC_BCDSEC_Write	写入秒时间数值
4	RTC_BCDSEC_Read	读取秒时间数值
5	RTC_BCDMIN_Write	写入分时间数值
6	RTC_BCDMIN_Read	读取分时间数值
7	RTC_BCDHOUR_Write	写入小时时间数值
8	RTC_BCDHOUR_Read	读取小时时间数值
9	RTC_BCDDATE_Write	写入天时间数值
10	RTC_BCDDATE_Read	读取天时间数值
11	RTC_BCDWEEK_Write	写入周时间数值
12	RTC_BCDWEEK_Read	读取周时间数值
13	RTC_BCDMONTH_Write	写入月时间数值
14	RTC_BCDMONTH_Read	读取月时间数值
15	RTC_BCDYEAR_Write	写入年时间数值
16	RTC_BCDYEAR_Read	读取年时间数值
17	RTC_MSECCNT_Write	设置毫秒计数器的数值
18	RTC_MSECCNT_Read	读取毫秒计数器的数值
19	RTC_ALARM_ALARMHOUR_Set	设置闹钟的小时数值
20	RTC_ALARM_ALARMHOUR_Get	获取闹钟的小时数值
21	RTC_ALARM_ALARMMIN_Set	设置闹钟的分数值
22	RTC_ALARM_ALARMMIN_Get	获取闹钟的分数值
23	RTC_ALARM_ALARMSEC_Set	设置闹钟的秒数值
24	RTC_ALARM_ALARMSEC_Get	获取闹钟的秒数值
25	RTC_FSEL_FSEL_Set	设置频率信号输出选择
26	RTC_FSEL_FSEL_Get	获取频率信号输出选择
27	RTC_STAMPEN_STAMP0EN_Setable	PB4 触发的时间戳功能使能
28	RTC_STAMPEN_STAMP0EN_Getable	读取 PB4 触发的时间戳功能使能状态
29	RTC_STAMPEN_STAMP1EN_Setable	PB5 触发的时间戳功能使能
30	RTC_STAMPEN_STAMP1EN_Getable	读取 PB5 触发的时间戳功能使能状态
31	RTC_CLKSTAMP0R_Read	读取 PB4 上升沿时间戳时分秒数据
32	RTC_CALSTAMP0R_Read	读取 PB4 上升沿时间戳年月周日数据
33	RTC_CLKSTAMP0F_Read	读取 PB4 下降沿时间戳时分秒数

		据
34	RTC_CALSTAMP0F_Read	读取 PB4 下降沿时间戳年月周日数据
35	RTC_CLKSTAMP1R_Read	读取 PB5 上升沿时间戳时分秒数据
36	RTC_CALSTAMP1R_Read	读取 PB5 上升沿时间戳年月周日数据
37	RTC_CLKSTAMP1F_Read	读取 PB5 下降沿时间戳时分秒数据
38	RTC_CALSTAMP1F_Read	读取 PB5 下降沿时间戳年月周日数据
39	RTC_RTCIE_SetableEx	设置 RTC 所需对应功能的中断使能
40	RTC_RTCIE_GetableEx	获取 RTC 所需对应功能的中断使能状态
41	RTC_RTCIF_ClrEx	清除 RTC 中断标志
42	RTC_RTCIF_ChkEx	读取 RTC 中断标志状态
43	RTC_TimeDate_SetEx	设置时间
44	RTC_TimeDate_GetEx	读取时间
45	RTC_AlarmTime_SetEx	设置闹钟时间
46	RTC_AlarmTime_GetEx	读取闹钟时间
47	RTC_CLKSTAMPxx_GetEx	读取时间戳数据
48	RTC_ADJUST_Write	设置时基计数器数值调整
49	RTC_ADJUST_Read	读取时基计数器数值
50	RTC_ADSIGN_ADSIGN_Set	设置时基计数器数值调整的方向调整
51	RTC_ADSIGN_ADSIGN_Get	读取时基计数器数值调整的方向调整
52	RTC_PR1SEN_PR1SEN_Setable	设置虚拟调校的功能使能
53	RTC_PR1SEN_PR1SEN_Getable	获取虚拟调校的功能使能状态
54	RTC_Trim_Proc	设置 RTC 时钟精度补偿

### 13.2.1 RTC\_Deinit

函数名	RTC_Deinit
函数原型	void RTC_Deinit(void)
功能描述	RTC 初始化为默认值

输入参数 1	无
输出参数	无
返回值	无

### 13.2.2 RTC\_RTCWE\_Write

函数名	RTC_RTCWE_Write
函数原型	void RTC_RTCWE_Write(uint32_t SetValue)
功能描述	写入 0xACACACAC, 允许 CPU 向 RTC 的 BCD 时间寄存器写入初值
输入参数 1	写入 0xACACACAC, 允许 CPU 向 RTC 的 BCD 时间寄存器写入初值, RTCWE 置 1 写入不为 0xACACACAC 任意值, 恢复写保护, RTCWE 清 0
输出参数	无
返回值	无

### 13.2.3 RTC\_BCDSEC\_Write

函数名	RTC_BCDSEC_Write
函数原型	void RTC_BCDSEC_Write(uint32_t SetValue)
功能描述	写入秒时间数值
输入参数 1	秒时间数值: 0-0X59 BCD 格式
输出参数	无
返回值	无

### 13.2.4 RTC\_BCDSEC\_Read

函数名	RTC_BCDSEC_Read
函数原型	uint32_t RTC_BCDSEC_Read(void)
功能描述	读取秒时间数值
输入参数 1	无
输出参数	无
返回值	秒时间输出: 0-0X59 BCD 格式

### 13.2.5 RTC\_BCDMIN\_Write

函数名	RTC_BCDMIN_Write
函数原型	void RTC_BCDMIN_Write(uint32_t SetValue)
功能描述	写入分时间数值
输入参数 1	分钟时间数值: 0-0X59 BCD 格式
输出参数	无

返回值	无
-----	---

### 13.2.6 RTC\_BCDMIN\_Read

函数名	RTC_BCDMIN_Read
函数原型	uint32_t RTC_BCDMIN_Read(void)
功能描述	读取分时间数值
输入参数 1	无
输出参数	无
返回值	分钟时间输出: 0-0X59 BCD 格式

### 13.2.7 RTC\_BCDHOUR\_Write

函数名	RTC_BCDHOUR_Write
函数原型	void RTC_BCDHOUR_Write(uint32_t SetValue)
功能描述	写入小时时间数值
输入参数 1	小时时间数值: 0-0X23 BCD 格式
输出参数	无
返回值	无

### 13.2.8 RTC\_BCDHOUR\_Read

函数名	RTC_BCDHOUR_Read
函数原型	uint32_t RTC_BCDHOUR_Read(void)
功能描述	读取小时时间数值
输入参数 1	无
输出参数	无
返回值	小时时间输出: 0-0X23 BCD 格式

### 13.2.9 RTC\_BCDDATE\_Write

函数名	RTC_BCDDATE_Write
函数原型	void RTC_BCDDATE_Write(uint32_t SetValue)
功能描述	写入天时间数值
输入参数 1	天时间数值: 0X01-0X31 BCD 格式
输出参数	无
返回值	无

**13.2.10 RTC\_BCDDATE\_Read**

函数名	RTC_BCDDATE_Read
函数原型	uint32_t RTC_BCDDATE_Read(void)
功能描述	读取天时间数值
输入参数 1	无
输出参数	无
返回值	天时间输出: 0X01-0X31 BCD 格式

**13.2.11 RTC\_BCDWEEK\_Write**

函数名	RTC_BCDWEEK_Write
函数原型	void RTC_BCDWEEK_Write(uint32_t SetValue)
功能描述	写入周时间数值
输入参数 1	周时间数值: 0-0X06 BCD 格式
输出参数	无
返回值	无

**13.2.12 RTC\_BCDWEEK\_Read**

函数名	RTC_BCDWEEK_Read
函数原型	uint32_t RTC_BCDWEEK_Read(void)
功能描述	读取周时间数值
输入参数 1	无
输出参数	无
返回值	周时间输出: 0-0X06 BCD 格式

**13.2.13 RTC\_BCDMONTH\_Write**

函数名	RTC_BCDMONTH_Write
函数原型	void RTC_BCDMONTH_Write(uint32_t SetValue)
功能描述	写入月时间数值
输入参数 1	月时间数值: 0X01-0X12 BCD 格式
输出参数	无
返回值	无

**13.2.14 RTC\_BCDMONTH\_Read**

函数名	RTC_BCDMONTH_Read
函数原型	uint32_t RTC_BCDMONTH_Read(void)
功能描述	读取月时间数值
输入参数 1	无
输出参数	无
返回值	月时间输出: 0X01-0X12 BCD 格式

**13.2.15 RTC\_BCDYEAR\_Write**

函数名	RTC_BCDYEAR_Write
函数原型	void RTC_BCDYEAR_Write(uint32_t SetValue)
功能描述	写入年时间数值
输入参数 1	年时间数值: 0-0X99 BCD 格式
输出参数	无
返回值	无

**13.2.16 RTC\_BCDYEAR\_Read**

函数名	RTC_BCDYEAR_Read
函数原型	uint32_t RTC_BCDYEAR_Read(void)
功能描述	读取年时间数值
输入参数 1	无
输出参数	无
返回值	年时间输出: 0-0X99 BCD 格式

**13.2.17 RTC\_MSECCNT\_Write**

函数名	RTC_MSECCNT_Write
函数原型	void RTC_MSECCNT_Write(uint32_t SetValue)
功能描述	设置毫秒计数器的数值
输入参数 1	毫秒计数器的数值(以 256Hz 为周期计数, 精度 3.9ms ): 0-255
输出参数	无
返回值	无



**13.2.18 RTC\_MSECCNT\_Read**

函数名	RTC_MSECCNT_Read
函数原型	uint32_t RTC_MSECCNT_Read(void)
功能描述	读取毫秒计数器的数值
输入参数 1	无
输出参数	无
返回值	毫秒计数器的数值(以 256Hz 为周期计数, 精度 3.9ms ): 0-255

**13.2.19 RTC\_ALARM\_ALARMHOUR\_Set**

函数名	RTC_ALARM_ALARMHOUR_Set
函数原型	void RTC_ALARM_ALARMHOUR_Set(uint32_t SetValue)
功能描述	设置闹钟的小时数值
输入参数 1	闹钟小时数据: 0-63
输出参数	无
返回值	无

**13.2.20 RTC\_ALARM\_ALARMHOUR\_Get**

函数名	RTC_ALARM_ALARMHOUR_Get
函数原型	uint32_t RTC_ALARM_ALARMHOUR_Get(void)
功能描述	获取闹钟的小时数值
输入参数 1	无
输出参数	无
返回值	闹钟小时数据: 0-63

**13.2.21 RTC\_ALARM\_ALARMMIN\_Set**

函数名	RTC_ALARM_ALARMMIN_Set
函数原型	void RTC_ALARM_ALARMMIN_Set(uint32_t SetValue)
功能描述	设置闹钟的分数值
输入参数 1	闹钟分钟数据: 0-127
输出参数	无
返回值	无

**13.2.22 RTC\_ALARM\_ALARMMIN\_Get**

函数名	RTC_ALARM_ALARMMIN_Get
函数原型	uint32_t RTC_ALARM_ALARMMIN_Get(void)
功能描述	获取闹钟的分数值
输入参数 1	无
输出参数	无
返回值	闹钟分钟数据：0-127

**13.2.23 RTC\_ALARM\_ALARMSEC\_Set**

函数名	RTC_ALARM_ALARMSEC_Set
函数原型	void RTC_ALARM_ALARMSEC_Set(uint32_t SetValue)
功能描述	设置闹钟的秒数值
输入参数 1	闹钟秒数据：0-127
输出参数	无
返回值	无

**13.2.24 RTC\_ALARM\_ALARMSEC\_Get**

函数名	RTC_ALARM_ALARMSEC_Get
函数原型	uint32_t RTC_ALARM_ALARMSEC_Get(void)
功能描述	获取闹钟的秒数值
输入参数 1	无
输出参数	无
返回值	闹钟秒数据：0-127

**13.2.25 RTC\_FSEL\_FSEL\_Set**

函数名	RTC_FSEL_FSEL_Set
函数原型	void RTC_FSEL_FSEL_Set(uint32_t SetValue)
功能描述	设置频率信号输出选择
输入参数 1	频率信号输出选择： 0000：输出 PLL 分频得到的精确 1 秒方波 0001：输出 PLL 分频的高电平宽度 80ms 的秒时标 0010：输出秒计数器进位信号，高电平宽度 1s 0011：输出分计数器进位信号，高电平宽度 1s 0100：输出小时计数器进位信号，高电平宽度 1s

	0101: 输出天计数器进位信号, 高电平宽度 1s 0110: 输出闹钟匹配信号 0111: 输出 128 秒方波信号 1000: 反向输出 PLL 分频的高电平宽度 80ms 的秒时标 1001: 反向输出秒计数器进位信号 1010: 反向输出分计数器进位信号 1011: 反向输出小时计数器进位信号 1100: 反向输出天计数器进位信号 1101: 反向输出闹钟匹配信号 1110: 反向输出 PLL 分频得到的精确 1 秒方波 1111: 输出 RTC 内部秒时标方波
输出参数	无
返回值	无

### 13.2.26 RTC\_FSEL\_FSEL\_Get

函数名	RTC_FSEL_FSEL_Get
函数原型	uint32_t RTC_FSEL_FSEL_Get(void)
功能描述	获取频率信号输出选择
输入参数 1	无
输出参数	无
返回值	频率信号输出选择: 0000: 输出 PLL 分频得到的精确 1 秒方波 0001: 输出 PLL 分频的高电平宽度 80ms 的秒时标 0010: 输出秒计数器进位信号, 高电平宽度 1s 0011: 输出分计数器进位信号, 高电平宽度 1s 0100: 输出小时计数器进位信号, 高电平宽度 1s 0101: 输出天计数器进位信号, 高电平宽度 1s 0110: 输出闹钟匹配信号 0111: 输出 128 秒方波信号 1000: 反向输出 PLL 分频的高电平宽度 80ms 的秒时标 1001: 反向输出秒计数器进位信号 1010: 反向输出分计数器进位信号 1011: 反向输出小时计数器进位信号 1100: 反向输出天计数器进位信号 1101: 反向输出闹钟匹配信号 1110: 反向输出 PLL 分频得到的精确 1 秒方波 1111: 输出 RTC 内部秒时标方波

**13.2.27 RTC\_STAMPEN\_STAMP0EN\_Setable**

函数名	RTC_STAMPEN_STAMP0EN_Setable
函数原型	void RTC_STAMPEN_STAMP0EN_Setable(FunState NewState)
功能描述	PB4 触发的时间戳功能使能
输入参数 1	PB4 触发的时间戳功能使能： DISABLE：禁止 PB4 触发的时间戳功能 ENABLE：使能 PB4 触发的时间戳功能
输出参数	无
返回值	无

**13.2.28 RTC\_STAMPEN\_STAMP0EN\_Getable**

函数名	RTC_STAMPEN_STAMP0EN_Getable
函数原型	FunState RTC_STAMPEN_STAMP0EN_Getable(void)
功能描述	读取 PB4 触发的时间戳功能使能状态
输入参数 1	无
输出参数	无
返回值	PB4 触发的时间戳功能使能： DISABLE：禁止 PB4 触发的时间戳功能 ENABLE：使能 PB4 触发的时间戳功能

**13.2.29 RTC\_STAMPEN\_STAMP1EN\_Setable**

函数名	RTC_STAMPEN_STAMP1EN_Setable
函数原型	void RTC_STAMPEN_STAMP1EN_Setable(FunState NewState)
功能描述	PB5 触发的时间戳功能使能
输入参数 1	PB5 触发的时间戳功能使能： DISABLE：禁止 PB4 触发的时间戳功能 ENABLE：使能 PB4 触发的时间戳功能
输出参数	无
返回值	无

**13.2.30 RTC\_STAMPEN\_STAMP1EN\_Getable**

函数名	RTC_STAMPEN_STAMP1EN_Getable
函数原型	FunState RTC_STAMPEN_STAMP1EN_Getable(void)

功能描述	读取 PB5 触发的时间戳功能使能状态
输入参数 1	无
输出参数	无
返回值	PB5 触发的时间戳功能使能： DISABLE：禁止 PB4 触发的时间戳功能 ENABLE：使能 PB4 触发的时间戳功能

### 13.2.31 RTC\_CLKSTAMP0R\_Read

函数名	RTC_CLKSTAMP0R_Read
函数原型	uint32_t RTC_CLKSTAMP0R_Read(void)
功能描述	读取 PB4 上升沿时间戳时分秒数据
输入参数 1	无
输出参数	无
返回值	PB4 上升沿后存储的 BCD 时分秒数据 time： time[6:0]:BCD 秒 time[14:8]:BCD 分 time[21:16]:BCD 时

### 13.2.32 RTC\_CALSTAMP0R\_Read

函数名	RTC_CALSTAMP0R_Read
函数原型	uint32_t RTC_CALSTAMP0R_Read(void)
功能描述	读取 PB4 上升沿时间戳年月周日数据
输入参数 1	无
输出参数	无
返回值	PB4 上升沿后存储的 BCD 年月周日数据 date： date [5:0]:BCD 天 date [10:8]:BCD 周 date [20:16]:BCD 月 date [31:24]:BCD 年

### 13.2.33 RTC\_CLKSTAMP0F\_Read

函数名	RTC_CLKSTAMP0F_Read
函数原型	uint32_t RTC_CLKSTAMP0F_Read(void)
功能描述	读取 PB4 下降沿时间戳时分秒数据
输入参数 1	无

输出参数	无
返回值	PB4 下降沿后存储的 BCD 时分秒数据 time: time[6:0]:BCD 秒 time[14:8]:BCD 分 time[21:16]:BCD 时

### 13.2.34 RTC\_CALSTAMP0F\_Read

函数名	RTC_CALSTAMP0F_Read
函数原型	uint32_t RTC_CALSTAMP0F_Read(void)
功能描述	读取 PB4 下降沿时间戳年月周日数据
输入参数 1	无
输出参数	无
返回值	PB4 下降沿后存储的 BCD 年月周日数据 date: date [5:0]:BCD 天 date [10:8]:BCD 周 date [20:16]:BCD 月 date [31:24]:BCD 年

### 13.2.35 RTC\_CLKSTAMP1R\_Read

函数名	RTC_CLKSTAMP1R_Read
函数原型	uint32_t RTC_CLKSTAMP1R_Read(void)
功能描述	读取 PB5 上升沿时间戳时分秒数据
输入参数 1	无
输出参数	无
返回值	PB5 上升沿后存储的 BCD 时分秒数据 time: time[6:0]:BCD 秒 time[14:8]:BCD 分 time[21:16]:BCD 时

### 13.2.36 RTC\_CALSTAMP1R\_Read

函数名	RTC_CALSTAMP1R_Read
函数原型	uint32_t RTC_CALSTAMP1R_Read(void)
功能描述	读取 PB5 上升沿时间戳年月周日数据
输入参数 1	无
输出参数	无



返回值	PB5 上升沿后存储的 BCD 年月周日数据 date: date [5:0]:BCD 天 date [10:8]:BCD 周 date [20:16]:BCD 月 date [31:24]:BCD 年
-----	-------------------------------------------------------------------------------------------------------------------

### 13.2.37 RTC\_CLKSTAMP1F\_Read

函数名	RTC_CLKSTAMP1F_Read
函数原型	uint32_t RTC_CLKSTAMP1F_Read(void)
功能描述	读取 PB5 下降沿时间戳时分秒数据
输入参数 1	无
输出参数	无
返回值	PB5 下降沿后存储的 BCD 时分秒数据 time: time[6:0]:BCD 秒 time[14:8]:BCD 分 time[21:16]:BCD 时

### 13.2.38 RTC\_CALSTAMP1F\_Read

函数名	RTC_CALSTAMP1F_Read
函数原型	uint32_t RTC_CALSTAMP1F_Read(void)
功能描述	读取 PB5 下降沿时间戳年月周日数据
输入参数 1	无
输出参数	无
返回值	PB5 下降沿后存储的 BCD 年月周日数据 date: date [5:0]:BCD 天 date [10:8]:BCD 周 date [20:16]:BCD 月 date [31:24]:BCD 年

### 13.2.39 RTC\_RTCIE\_SetableEx

函数名	RTC_RTCIE_SetableEx
函数原型	void RTC_RTCIE_SetableEx(FunState NewState, uint32_t ie_def)
功能描述	设置 RTC 所需对应功能的中断使能
输入参数 1	RTC 使能状态: DISABLE: 禁止 RTC

	ENABLE: 使能 RTC
输入参数 2	RTC 所需对应功能的中断使能
输出参数	无
返回值	无

### 13.2.40 RTC\_RTCIE\_GetableEx

函数名	RTC_RTCIE_GetableEx
函数原型	FunState RTC_RTCIE_GetableEx(uint32_t ie_def)
功能描述	获取 RTC 所需对应功能的中断使能状态
输入参数 1	RTC 中的一或多个中断
输出参数	无
返回值	RTC 中的某个中断使能状态: DISABLE: 禁止 RTC 中的一或多个中断 ENABLE: 使能 RTC 中的一或多个中断

### 13.2.41 RTC\_RTCIF\_ClrEx

函数名	RTC_RTCIF_ClrEx
函数原型	void RTC_RTCIF_ClrEx(uint32_t if_def)
功能描述	清除 RTC 中断标志
输入参数 1	要清除的所使用的 RTC 中断的中断标志
输出参数	无
返回值	无

### 13.2.42 RTC\_RTCIF\_ChkEx

函数名	RTC_RTCIF_ChkEx
函数原型	FlagStatus RTC_RTCIF_ChkEx(uint32_t if_def)
功能描述	读取所使用的 RTC 中断标志状态
输入参数 1	RTC 中的一或多个中断
输出参数	无
返回值	所使用的 RTC 中断标志状态

### 13.2.43 RTC\_TimeDate\_SetEx

函数名	RTC_TimeDate_SetEx
函数原型	void RTC_TimeDate_SetEx(RTC_TimeDateTypeDef* para)



功能描述	设置时间							
输入参数 1	<div>RTC 时间参数:</div> <table><tr><td>RTC 年: 0-0X99 BCD 格式</td></tr><tr><td>RTC 月: 0X01-0X12 BCD 格式</td></tr><tr><td>RTC 日: 0X01-0X31 BCD 格式</td></tr><tr><td>RTC 小时: 0-0X23 BCD 格式</td></tr><tr><td>RTC 分钟: 0-0X59 BCD 格式</td></tr><tr><td>RTC 秒: 0-0X59 BCD 格式</td></tr><tr><td>RTC 周: 0-0X06 BCD 格式</td></tr></table>	RTC 年: 0-0X99 BCD 格式	RTC 月: 0X01-0X12 BCD 格式	RTC 日: 0X01-0X31 BCD 格式	RTC 小时: 0-0X23 BCD 格式	RTC 分钟: 0-0X59 BCD 格式	RTC 秒: 0-0X59 BCD 格式	RTC 周: 0-0X06 BCD 格式
RTC 年: 0-0X99 BCD 格式								
RTC 月: 0X01-0X12 BCD 格式								
RTC 日: 0X01-0X31 BCD 格式								
RTC 小时: 0-0X23 BCD 格式								
RTC 分钟: 0-0X59 BCD 格式								
RTC 秒: 0-0X59 BCD 格式								
RTC 周: 0-0X06 BCD 格式								
输出参数	无							
返回值	无							

### 13.2.44 RTC\_TimeDate\_GetEx

函数名	RTC_TimeDate_GetEx							
函数原型	void RTC_TimeDate_GetEx(RTC_TimeDateTypeDef* para)							
功能描述	读取时间							
输入参数 1	无							
输出参数	<div>RTC 时间参数：</div> <table><tr><td>RTC 年：0-0X99 BCD 格式</td></tr><tr><td>RTC 月：0X01-0X12 BCD 格式</td></tr><tr><td>RTC 日：0X01-0X31 BCD 格式</td></tr><tr><td>RTC 小时：0-0X23 BCD 格式</td></tr><tr><td>RTC 分钟：0-0X59 BCD 格式</td></tr><tr><td>RTC 秒：0-0X59 BCD 格式</td></tr><tr><td>RTC 周：0-0X06 BCD 格式</td></tr></table>	RTC 年：0-0X99 BCD 格式	RTC 月：0X01-0X12 BCD 格式	RTC 日：0X01-0X31 BCD 格式	RTC 小时：0-0X23 BCD 格式	RTC 分钟：0-0X59 BCD 格式	RTC 秒：0-0X59 BCD 格式	RTC 周：0-0X06 BCD 格式
RTC 年：0-0X99 BCD 格式								
RTC 月：0X01-0X12 BCD 格式								
RTC 日：0X01-0X31 BCD 格式								
RTC 小时：0-0X23 BCD 格式								
RTC 分钟：0-0X59 BCD 格式								
RTC 秒：0-0X59 BCD 格式								
RTC 周：0-0X06 BCD 格式								
返回值	无							

### 13.2.45 RTC\_AlarmTime\_SetEx

函数名	RTC_AlarmTime_SetEx			
函数原型	void RTC_AlarmTime_SetEx(RTC_AlarmTmieTypeDef* para)			
功能描述	设置闹钟时间			
输入参数 1	<div>闹钟时间参数：<table><tr><td>RTC 闹钟时: 0-63</td></tr><tr><td>RTC 闹钟分:0-127</td></tr><tr><td>RTC 闹钟秒:0-127</td></tr></table></div>	RTC 闹钟时: 0-63	RTC 闹钟分:0-127	RTC 闹钟秒:0-127
RTC 闹钟时: 0-63				
RTC 闹钟分:0-127				
RTC 闹钟秒:0-127				
输出参数	无			
返回值	无			

**13.2.46 RTC\_AlarmTime\_GetEx**

函数名	RTC_AlarmTime_GetEx			
函数原型	void RTC_AlarmTime_GetEx(RTC_AlarmTmieTypeDef* para)			
功能描述	读取闹钟时间			
输入参数 1	无			
输出参数	<div>闹钟时间参数：<table><tr><td>RTC 闹钟时： 0-63</td></tr><tr><td>RTC 闹钟分:0-127</td></tr><tr><td>RTC 闹钟秒:0-127</td></tr></table></div>	RTC 闹钟时： 0-63	RTC 闹钟分:0-127	RTC 闹钟秒:0-127
RTC 闹钟时： 0-63				
RTC 闹钟分:0-127				
RTC 闹钟秒:0-127				
返回值	无			

**13.2.47 RTC\_CLKSTAMPxx\_GetEx**

函数名	RTC_CLKSTAMPxx_GetEx							
函数原型	void RTC_CLKSTAMPxx_GetEx(uint8_t StampType_def, RTC_StampTmieTypeDef* para)							
功能描述	读取时间戳数据							
输入参数 1	时间戳选择： 0：上升沿时间戳 0 1：下降沿时间戳 0 2：上升沿时间戳 1 3：下降沿时间戳 1							
输出参数	时间戳数据： <table><tr><td>RTC 年：0-0X99 BCD 格式</td></tr><tr><td>RTC 月：0X01-0X12 BCD 格式</td></tr><tr><td>RTC 日：0X01-0X31 BCD 格式</td></tr><tr><td>RTC 小时：0-0X23 BCD 格式</td></tr><tr><td>RTC 分钟：0-0X59 BCD 格式</td></tr><tr><td>RTC 秒：0-0X59 BCD 格式</td></tr><tr><td>RTC 周：0-0X06 BCD 格式</td></tr></table>	RTC 年：0-0X99 BCD 格式	RTC 月：0X01-0X12 BCD 格式	RTC 日：0X01-0X31 BCD 格式	RTC 小时：0-0X23 BCD 格式	RTC 分钟：0-0X59 BCD 格式	RTC 秒：0-0X59 BCD 格式	RTC 周：0-0X06 BCD 格式
RTC 年：0-0X99 BCD 格式								
RTC 月：0X01-0X12 BCD 格式								
RTC 日：0X01-0X31 BCD 格式								
RTC 小时：0-0X23 BCD 格式								
RTC 分钟：0-0X59 BCD 格式								
RTC 秒：0-0X59 BCD 格式								
RTC 周：0-0X06 BCD 格式								
返回值	无							

**13.2.48 RTC\_ADJUST\_Write**

函数名	RTC_ADJUST_Write
函数原型	void RTC_ADJUST_Write(uint32_t SetValue)
功能描述	设置时基计数器数值调整
输入参数 1	时基计数器数值调整值: 0-7FF



输出参数	无
返回值	无

### 13.2.49 RTC\_ADJUST\_Read

函数名	RTC_ADJUST_Read
函数原型	uint32_t RTC_ADJUST_Read(void)
功能描述	读取时基计数器数值
输入参数 1	无
输出参数	无
返回值	时基计数器数值调整值：0-7FF

### 13.2.50 RTC\_ADSIGN\_ADSIGN\_Set

函数名	RTC_ADSIGN_ADSIGN_Set
函数原型	void RTC_ADSIGN_ADSIGN_Set(uint32_t SetValue)
功能描述	设置时基计数器数值调整的方向调整
输入参数 1	时基计数器数值调整的方向调整： 0：表示增加计数初值 1：表示减少计数初值
输出参数	无
返回值	无

### 13.2.51 RTC\_ADSIGN\_ADSIGN\_Get

函数名	RTC_ADSIGN_ADSIGN_Get
函数原型	uint32_t RTC_ADSIGN_ADSIGN_Get(void)
功能描述	读取时基计数器数值调整的方向调整
输入参数 1	无
输出参数	无
返回值	时基计数器数值调整的方向调整： 0：表示增加计数初值 1：表示减少计数初值

### 13.2.52 RTC\_PR1SEN\_PR1SEN\_Setable

函数名	RTC_PR1SEN_PR1SEN_Setable
函数原型	void RTC_PR1SEN_PR1SEN_Setable(FunState NewState)

功能描述	设置虚拟调校的功能使能
输入参数 1	虚拟调校的功能使能： DISABLE：禁止虚拟调校功能 ENABLE：使能虚拟调校功能
输出参数	无
返回值	无

### 13.2.53 RTC\_PR1SEN\_PR1SEN\_Getable

函数名	RTC_PR1SEN_PR1SEN_Getable
函数原型	FunState RTC_PR1SEN_PR1SEN_Getable(void)
功能描述	获取虚拟调校的功能使能状态
输入参数 1	无
输出参数	无
返回值	虚拟调校的功能使能： DISABLE：禁止虚拟调校功能 ENABLE：使能虚拟调校功能

### 13.2.54 RTC\_Trim\_Proc

函数名	RTC_Trim_Proc
函数原型	void RTC_Trim_Proc(float err_ppm)
功能描述	设置 RTC 时钟精度补偿
输入参数 1	float 型参误差 ppm
输出参数	无
返回值	无

# 14 AES

## 14.1 AES 寄存器结构

寄存器	描述
AESCR	AES 控制寄存器
AESIF	AES 中断标志寄存器
AESDIN	AES 数据输入寄存器
AESDOUT	AES 数据输出寄存器
AESKEY0	AES 密钥寄存器 0
AESKEY1	AES 密钥寄存器 1
AESKEY2	AES 密钥寄存器 2
AESKEY3	AES 密钥寄存器 3
AESKEY4	AES 密钥寄存器 4
AESKEY5	AES 密钥寄存器 5
AESKEY6	AES 密钥寄存器 6
AESKEY7	AES 密钥寄存器 7
AESIVR0	AES 初始向量寄存器 0
AESIVR1	AES 初始向量寄存器 1
AESIVR2	AES 初始向量寄存器 2
AESIVR3	AES 初始向量寄存器 3

## 14.2 AES 库函数

序号	函数名	描述
1	AES_Init	AES 初始化
2	AES_Deinit	AES 初始化为缺省值
3	AES_AESCR_AESEN_Setable	设置 AES 功能使能
4	AES_AESCR_AESEN_Getable	获取 AES 功能使能设置
5	AES_AESCR_MODE_Set	设置 AES 工作模式
6	AES_AESCR_MODE_Get	获取 AES 工作模式
7	AES_AESCR_CHMOD_Set	设置 AES 数据流处理模式
8	AES_AESCR_CHMOD_Get	获取 AES 数据流处理模式
9	AES_AESCR_DATATYP_Set	设置 AES 数据类型
10	AES_AESCR_DATATYP_Get	获取 AES 数据类型设置

11	AES_AESDIN_Write	设置 AES 数据输入值
12	AES_AESDIN_Read	获取 AES 数据输入值
13	AES_AESDOUT_Write	设置 AES 数据输出值
14	AES_AESDOUT_Read	获取 AES 数据输出值
15	AES_AESCR_KEYLEN_Set	设置 AES 加密密钥长度
16	AES_AESCR_KEYLEN_Get	获取 AES 加密密钥长度设置
17	AES_AESDIN_GroupWrite_128BIT	设置加解密数据输入
18	AES_AESDOUT_GroupRead_128BIT	获取加解密数据输出
19	AES_GroupWriteAndRead_128BIT	加解密数据的写与读
20	AES_AESKEY_WriteEx	设置加解密密钥输入(key0 存放密钥最低 32bit)
21	AES_AESKEY_ReadEx	获取加解密密钥输出(key0 存放密钥最低 32bit)
22	AES_AESIVR_WriteEx	设置初始向量输入
23	AES_AESIVR_ReadEx	获取初始向量
24	AES_AESCR_ERRIE_Setable	设置错误标志 (RDERR 和 WRERR) 中断使能
25	AES_AESCR_ERRIE_Getable	获取错误标志 (RDERR 和 WRERR) 中断使能设置
26	AES_AESCR_CCFIE_Setable	设置 CCF 标志中断使能
27	AES_AESCR_CCFIE_Getable	获取 CCF 标志中断使能设置
28	AES_AESCR_ERRC_Clr	清除错误标志
29	AES_AESCR_CCFC_Clr	清除 CCF 标志
30	AES_AESIF_WRERR_Chk	获取 AES 写错误标志
31	AES_AESIF_RDERR_Chk	获取 AES 读错误标志
32	AES_AESIF_CCF_Chk	获取 AES 计算完成标志
33	AES_AESCR_DMAIEN_Setable	设置 DMA 自动写数据使能
34	AES_AESCR_DMAIEN_Getable	获取 DMA 自动写数据使能设置
35	AES_AESCR_DMAOEN_Setable	设置 DMA 自动读数据使能
36	AES_AESCR_DMAOEN_Getable	获取 DMA 自动读数据使能设置

### 14.2.1 AES\_Init

函数名	AES_Init
函数原型	void AES_Init(AES_InitTypeDef* para)
功能描述	AES 模块初始化函数
输入参数 1	AES 参数配置:

	<p>AES 加密密钥长度, AESEN=1 时不可修改:</p> <p>00: 128bit</p> <p>01: 192bit</p> <p>10: 256bit</p> <p>11: 保留</p>
	<p>AES 数据流处理模式, AESEN=1 时不可修改:</p> <p>00: ECB</p> <p>01: CBC</p> <p>10: CTR</p> <p>11: 使用 MultH 模块</p>
	<p>AES 工作模式, AESEN=1 时不可修改:</p> <p>00: 加密</p> <p>01: 密钥扩展</p> <p>10: 解密</p> <p>11: 密钥扩展+解密</p>
	<p>选择数据类型, AESEN=1 时不可修改:</p> <p>00: 32bit 数据不交换</p> <p>01: 16bit 数据半字交换</p> <p>10: 8bit 数据字节交换</p> <p>11: 1bit 数据比特交换</p>
	<p>DMA 数据自动读出使能:</p> <p>DISABLE: 禁止 DMA 数据自动读出</p> <p>ENABLE: 使能 DMA 数据自动读出</p>
	<p>DMA 数据自动写入使能:</p> <p>DISABLE: 禁止 DMA 数据自动写入</p> <p>ENABLE: 使能 DMA 数据自动写入</p>
	<p>错误标志 (RDERR 和 WRERR) 中断使能:</p> <p>DISABLE: 禁止错误标志 (RDERR 和 WRERR) 中断</p> <p>ENABLE: 使能错误标志 (RDERR 和 WRERR) 中断</p>
	<p>CCF 标志中断使能:</p> <p>DISABLE: 禁止 CCF 标志中断</p> <p>ENABLE: 使能 CCF 标志中断</p>
	<p>AES 使能:</p> <p>DISABLE: 禁止 AES</p> <p>ENABLE: 使能 AES</p>
输出参数	无
返回值	无

### 14.2.2 AES\_Deinit

函数名	AES_Deinit
函数原型	void AES_Deinit(void)

功能描述	AES 初始化为默认值
输入参数 1	无
输出参数	无
返回值	无

### 14.2.3 AES\_AESCR\_AESEN\_Setable

函数名	AES_AESCR_AESEN_Setable
函数原型	void AES_AESCR_AESEN_Setable(FunState NewState)
功能描述	配置 AES 功能使能
输入参数 1	AES 功能使能： DISABLE: 禁止 AES 功能 ENABLE: 使能 AES 功能
输出参数	无
返回值	无

### 14.2.4 AES\_AESCR\_AESEN\_Getable

函数名	AES_AESCR_AESEN_Getable
函数原型	FunState AES_AESCR_AESEN_Getable(void)
功能描述	获取 AES 功能使能配置
输入参数 1	无
输出参数	无
返回值	AES 功能使能： DISABLE: 禁止 AES 功能 ENABLE: 使能 AES 功能

### 14.2.5 AES\_AESCR\_MODE\_Set

函数名	AES_AESCR_MODE_Set
函数原型	void AES_AESCR_MODE_Set(uint32_t SetValue)
功能描述	配置 AES 工作模式
输入参数 1	AES 工作模式： AES_AESCR_MODE_ENCRYPT: 加密 AES_AESCR_MODE_KEYEXP: 密钥扩展 AES_AESCR_MODE_DECRYPT: 解密 AES_AESCR_MODE_KEYEXPDECRYPT: 密钥扩展+解密
输出参数	无



返回值	无
-----	---

#### 14.2.6 AES\_AESCR\_MODE\_Get

函数名	AES_AESCR_MODE_Get
函数原型	uint32_t AES_AESCR_MODE_Get(void)
功能描述	获取 AES 工作模式
输入参数 1	无
输出参数	无
返回值	AES 工作模式： AES_AESCR_MODE_ENCRYPT: 加密 AES_AESCR_MODE_KEYEXP: 密钥扩展 AES_AESCR_MODE_DECRYPT: 解密 AES_AESCR_MODE_KEYEXPDECRYPT: 密钥扩展+解密

#### 14.2.7 AES\_AESCR\_CHMOD\_Set

函数名	AES_AESCR_CHMOD_Set
函数原型	void AES_AESCR_CHMOD_Set(uint32_t SetValue)
功能描述	配置 AES 数据流处理模式(AESEN=0 时有效)
输入参数 1	AES 数据流处理模式： AES_AESCR_CHMOD_ECB: ECB AES_AESCR_CHMOD_CBC: CBC AES_AESCR_CHMOD_CTR: CTR AES_AESCR_CHMOD_MULTH: 使用 MultH 模块
输出参数	无
返回值	无

#### 14.2.8 AES\_AESCR\_CHMOD\_Get

函数名	AES_AESCR_CHMOD_Get
函数原型	uint32_t AES_AESCR_CHMOD_Get(void)
功能描述	获取 AES 数据流处理模式
输入参数 1	无
输出参数	无
返回值	AES 数据流处理模式： AES_AESCR_CHMOD_ECB: ECB AES_AESCR_CHMOD_CBC: CBC

	AES_AESCR_CHMOD_CTR: CTR AES_AESCR_CHMOD_MULTH: 使用 MultH 模块
--	----------------------------------------------------------------

### 14.2.9 AES\_AESCR\_DATATYP\_Set

函数名	AES_AESCR_DATATYP_Set
函数原型	void AES_AESCR_DATATYP_Set(uint32_t SetValue)
功能描述	配置 AES 数据类型(AESEN=0 时有效)
输入参数 1	AES 数据类型: AES_AESCR_DATATYP_32BITNOEX: 32bit 数据不交换 AES_AESCR_DATATYP_16BITEX: 16bit 数据半字交换 AES_AESCR_DATATYP_8BITEX: 8bit 数据字节交换 AES_AESCR_DATATYP_1BITEX: 1bit 数据比特交换
输出参数	无
返回值	无

### 14.2.10 AES\_AESCR\_DATATYP\_Get

函数名	AES_AESCR_DATATYP_Get
函数原型	uint32_t AES_AESCR_DATATYP_Get(void)
功能描述	获取 AES 数据类型配置
输入参数 1	无
输出参数	无
返回值	AES 数据类型: AES_AESCR_DATATYP_32BITNOEX: 32bit 数据不交换 AES_AESCR_DATATYP_16BITEX: 16bit 数据半字交换 AES_AESCR_DATATYP_8BITEX: 8bit 数据字节交换 AES_AESCR_DATATYP_1BITEX: 1bit 数据比特交换

### 14.2.11 AES\_AESDIN\_Write

函数名	AES_AESDIN_Write
函数原型	void AES_AESDIN_Write(uint32_t SetValue)
功能描述	配置 AES 数据输入寄存器的值
输入参数 1	AES 数据输入寄存器的值, 当 AES 需要输入加解密数据时, 应该往该寄存器连续写 4 次 模式 1 (加密): 把明文从 MSB 到 LSB 分 4 次写入。 模式 2 (密钥扩展): 无需使用数据输入寄存器 模式 3 和模式 4 (解密): 把密文从 MSB 到 LSB 分 4 次写入。

	MultH 模式：把乘数 A 或 C 从 MSB 到 LSB 分 4 次写入
输出参数	无
返回值	无

#### 14.2.12 AES\_AESDIN\_Read

函数名	AES_AESDIN_Read
函数原型	uint32_t AES_AESDIN_Read(void)
功能描述	获取 AES 数据输入寄存器的值
输入参数 1	无
输出参数	无
返回值	AES 数据寄存器值：0-FFFFFFFF

#### 14.2.13 AES\_AESDOUT\_Write

函数名	AES_AESDOUT_Write
函数原型	void AES_AESDOUT_Write(uint32_t SetValue)
功能描述	配置 AES 数据输出值
输入参数 1	AES 数据输出寄存器的值：0-FFFFFFFF
输出参数	无
返回值	无

#### 14.2.14 AES\_AESDOUT\_Read

函数名	AES_AESDOUT_Read
函数原型	uint32_t AES_AESDOUT_Read(void)
功能描述	获取 AES 数据输出寄存器
输入参数 1	无
输出参数	无
返回值	AES 数据输出寄存器数据： 模式 1（加密）：把密文从 MSB 到 LSB 分 4 次读出。 模式 2（密钥扩展）：无需使用数据输出寄存器 模式 3 和模式 4（解密）：把明文从 MSB 到 LSB 分 4 次输出。 MultH 模式：运算结果存储在 IVR 寄存器中，无需读取 AES_DOUTR 寄存器

#### 14.2.15 AES\_AESCR\_KEYLEN\_Set

函数名	AES_AESCR_KEYLEN_Set
-----	----------------------

函数原型	void AES_AESCR_KEYLEN_Set(uint32_t SetValue)
功能描述	配置 AES 加密密钥长度
输入参数 1	AES 加密密钥长度: AES_AESCR_KEYLEN_128BIT: 128bit AES_AESCR_KEYLEN_192BIT: 192bit AES_AESCR_KEYLEN_256BIT: 256bit
输出参数	无
返回值	无

#### 14.2.16 AES\_AESCR\_KEYLEN\_Get

函数名	AES_AESCR_KEYLEN_Get
函数原型	uint32_t AES_AESCR_KEYLEN_Get(void)
功能描述	获取 AES 加密密钥长度配置
输入参数 1	无
输出参数	无
返回值	AES 加密密钥长度: AES_AESCR_KEYLEN_128BIT: 128bit AES_AESCR_KEYLEN_192BIT: 192bit AES_AESCR_KEYLEN_256BIT: 256bit

#### 14.2.17 AES\_AESDIN\_GroupWrite\_128BIT

函数名	AES_AESDIN_GroupWrite_128BIT
函数原型	void AES_AESDIN_GroupWrite_128BIT(uint8_t *DataIn)
功能描述	配置加解密数据输入, 16 字节 (128bit) 分组数据输入
输入参数 1	待加解密的数据: 16 字节分组数据
输出参数	无
返回值	无

#### 14.2.18 AES\_AESDOUT\_GroupRead\_128BIT

函数名	AES_AESDOUT_GroupRead_128BIT
函数原型	void AES_AESDOUT_GroupRead_128BIT(uint8_t *DataOut)
功能描述	获取加解密数据输出, 16 字节 (128bit) 分组数据输出
输入参数 1	无
输出参数	已经完成加解密的数据: 16 字节分组数据
返回值	无

### 14.2.19 AES\_GroupWriteAndRead\_128BIT

函数名	AES_GroupWriteAndRead_128BIT
函数原型	uint8_t AES_GroupWriteAndRead_128BIT(uint8_t *DataIn, uint8_t *DataOut)
功能描述	加解密数据的写与读
输入参数 1	待加解密的数据: 16 字节分组数据
输出参数	已经完成加解密的数据: 16 字节分组数据
返回值	无

### 14.2.20 AES\_AESKEY\_WriteEx

函数名	AES_AESKEY_WriteEx
函数原型	void AES_AESKEY_WriteEx(uint8_t *KeyIn, uint8_t Len)
功能描述	配置加解密密钥输入(key0 存放密钥最低 32bit)
输入参数 1	加解密密钥输入: KeyIn: 密钥 16/24/32 字节长度的密钥
输入参数 2	Len: 密钥长度 16/24/32
输出参数	无
返回值	无

### 14.2.21 AES\_AESKEY\_ReadEx

函数名	AES_AESKEY_ReadEx
函数原型	void AES_AESKEY_ReadEx(uint8_t *KeyOut, uint8_t Len)
功能描述	加解密密钥输出(key0 存放密钥最低 32bit)
输入参数 1	Len: 密钥长度
输出参数	KeyOut: 加解密密钥
返回值	无

### 14.2.22 AES\_AESIVR\_WriteEx

函数名	AES_AESIVR_WriteEx
函数原型	void AES_AESIVR_WriteEx(uint8_t *IVRIn)
功能描述	设置初始向量输入
输入参数 1	输入的初始向量
输出参数	无
返回值	无

### 14.2.23 AES\_AESIVR\_ReadEx

函数名	AES_AESIVR_ReadEx
函数原型	void AES_AESIVR_ReadEx(uint8_t *IVROut)
功能描述	获取初始向量
输入参数 1	无
输出参数	输出的初始向量
返回值	无

### 14.2.24 AES\_AESCR\_ERRIE\_Setable

函数名	AES_AESCR_ERRIE_Setable
函数原型	void AES_AESCR_ERRIE_Setable(FunState NewState)
功能描述	配置错误标志 (RDERR 和 WRERR) 中断使能
输入参数 1	错误标志 (RDERR 和 WRERR) 中断使能: DISABLE: 禁止错误标志 (RDERR 和 WRERR) 中断 ENABLE: 使能错误标志 (RDERR 和 WRERR) 中断
输出参数	无
返回值	无

### 14.2.25 AES\_AESCR\_ERRIE\_Getable

函数名	AES_AESCR_ERRIE_Getable
函数原型	FunState AES_AESCR_ERRIE_Getable(void)
功能描述	获取错误标志 (RDERR 和 WRERR) 中断使能配置
输入参数 1	无
输出参数	无
返回值	错误标志 (RDERR 和 WRERR) 中断使能: DISABLE: 禁止错误标志 (RDERR 和 WRERR) 中断 ENABLE: 使能错误标志 (RDERR 和 WRERR) 中断

### 14.2.26 AES\_AESCR\_CCFIE\_Setable

函数名	AES_AESCR_CCFIE_Setable
函数原型	void AES_AESCR_CCFIE_Setable(FunState NewState)
功能描述	配置 CCF 标志中断使能
输入参数 1	CCF 标志中断使能:

	DISABLE: 禁止 CCF 标志中断 ENABLE: 使能 CCF 标志中断
输出参数	无
返回值	无

#### 14.2.27 AES\_AESCR\_CCFIE\_Getable

函数名	AES_AESCR_CCFIE_Getable
函数原型	FunState AES_AESCR_CCFIE_Getable(void)
功能描述	获取 CCF 标志中断使能配置
输入参数 1	无
输出参数	无
返回值	CCF 标志中断使能: DISABLE: 禁止 CCF 标志中断 ENABLE: 使能 CCF 标志中断

#### 14.2.28 AES\_AESCR\_ERRC\_Clr

函数名	AES_AESCR_ERRC_Clr
函数原型	void AES_AESCR_ERRC_Clr(void)
功能描述	清除错误标志
输入参数 1	无
输出参数	无
返回值	无

#### 14.2.29 AES\_AESCR\_CCFC\_Clr

函数名	AES_AESCR_CCFC_Clr
函数原型	void AES_AESCR_CCFC_Clr(void)
功能描述	清除 CCF 标志
输入参数 1	无
输出参数	无
返回值	无

#### 14.2.30 AES\_AESIF\_WRERR\_Chk

函数名	AES_AESIF_WRERR_Chk
函数原型	FlagStatus AES_AESIF_WRERR_Chk(void)

功能描述	获取 AES 写错误标志
输入参数 1	无
输出参数	无
返回值	AES 写错误标志： RESET：计算和输出阶段未发生写操作 SET：计算或输出阶段发生写操作

#### 14.2.31 AES\_AESIF\_RDERR\_Chk

函数名	AES_AESIF_RDERR_Chk
函数原型	FlagStatus AES_AESIF_RDERR_Chk(void)
功能描述	获取 AES 读错误标志
输入参数 1	无
输出参数	无
返回值	AES 读错误标志： RESET：计算和输出阶段未发生读操作 SET：计算或输出阶段发生读操作

#### 14.2.32 AES\_AESIF\_CCF\_Chk

函数名	AES_AESIF_CCF_Chk
函数原型	FlagStatus AES_AESIF_CCF_Chk(void)
功能描述	获取 AES 计算完成标志
输入参数 1	无
输出参数	无
返回值	AES 计算完成标志： RESET：计算没有完成 SET：计算完成

#### 14.2.33 AES\_AESCR\_DMAIEN\_Setable

函数名	AES_AESCR_DMAIEN_Setable
函数原型	void AES_AESCR_DMAIEN_Setable(FunState NewState)
功能描述	配置 DMA 自动写数据使能(工作模式 1,3,4 有效)
输入参数 1	DMA 自动写数据使能： DISABLE：禁止 DMA 自动写数据 ENABLE：使能 DMA 自动写数据
输出参数	无





返回值	无
-----	---

#### 14.2.34 AES\_AESCR\_DMAIEN\_Getable

函数名	AES_AESCR_DMAIEN_Getable
函数原型	FunState AES_AESCR_DMAIEN_Getable(void)
功能描述	获取 DMA 自动写数据使能配置(工作模式 1,3,4 有效)
输入参数 1	无
输出参数	无
返回值	DMA 自动写数据使能: DISABLE: 禁止 DMA 自动写数据 ENABLE: 使能 DMA 自动写数据

#### 14.2.35 AES\_AESCR\_DMAOEN\_Setable

函数名	AES_AESCR_DMAOEN_Setable
函数原型	void AES_AESCR_DMAOEN_Setable(FunState NewState)
功能描述	配置 DMA 自动读数据使能
输入参数 1	DMA 自动读数据使能: DISABLE: 禁止 DMA 自动读数据 ENABLE : 使能 DMA 自动读数据
输出参数	无
返回值	无

#### 14.2.36 AES\_AESCR\_DMAOEN\_Getable

函数名	AES_AESCR_DMAOEN_Getable
函数原型	FunState AES_AESCR_DMAOEN_Getable(void)
功能描述	获取 DMA 自动读数据使能配置(工作模式 1,3,4 有效)
输入参数 1	无
输出参数	无
返回值	DMA 自动读数据使能: DISABLE: 禁止 DMA 自动读数据 ENABLE : 使能 DMA 自动读数据

# CRC

## CRC 寄存器结构

寄存器	描述
CRCDR	CRC 数据寄存器
CRCCR	CRC 控制寄存器
CRCLFSR	CRC 运算寄存器
CRCXOR	CRC 异或输出寄存器
CRCFLSEN	CRC Flash 校验寄存器
CRCFLSAD	CRC Flash 校验起始地址寄存器
CRCFLSSIZE	CRC Flash 校验数据长度寄存器

## CRC 库函数

序号	函数名	描述
1	CRC_Init	CRC 初始化
2	CRC_Deinit	CRC 初始化为缺省值
3	CRC_CRCDR_Write	填充 CRC 数据输入
4	CRC_CRCDR_Read	获取 CRC 数据输入
5	CRC_CRCCR_RFLTIN_Setable	设置 CRC 输入反转
6	CRC_CRCCR_RFLTIN_Getable	获取 CRC 输入反转设置
7	CRC_CRCCR_RFLTOUT_Setable	设置 CRC 输出反转
8	CRC_CRCCR_RFLTOUT_Getable	获取 CRC 输出反转设置
9	CRC_CRCCR_RES_Chk	获取 CRC 结果标志位
10	CRC_CRCCR_BUSY_Chk	获取 CRC 运算标志位
11	CRC_CRCCR_XOR_Setable	设置 CRC 输出异或使能
12	CRC_CRCCR_XOR_Getable	获取 CRC 输出异或设置
13	CRC_CRCCR_CRCSEL_Set	设置 CRC 校验多项式选择
14	CRC_CRCCR_CRCSEL_Get	获取 CRC 校验多项式选择
15	CRC_CRCLFSR_Write	设置 CRC 运算初值
16	CRC_CRCLFSR_Read	获取 CRC 运算初值
17	CRC_CRCXOR_Write	设置 CRC 输出异或值
18	CRC_CRCXOR_Read	获取 CRC 输出异或值
19	CRC_CRCFLSEN_FLSCRCEN_Setable	设置 CRC 校验使能



20	CRC_CRCFLSEN_FLSCRCEN_Getable	获取 CRC 校验使能设置
21	CRC_CRCFLSAD_Write	设置 FLASH CRC 校验起始地址
22	CRC_CRCFLSAD_Read	获取 FLASH CRC 校验起始地址
23	CRC_CRCFLSSIZE_Write	设置 FLASH CRC 校验长度
24	CRC_CRCFLSSIZE_Read	获取 FLASH CRC 校验长度

### 15.2.1 CRC\_Init

函数名	CRC_Init					
函数原型	void CRC_Init(CRC_InitTypeDef* para)					
功能描述	CRC 初始化					
输入参数 1	<div>CRC 参数：</div> <table><tr><td>CRC 校验多项式选择： 00: CRC16 01: CRC8 10/11: CCITT</td></tr><tr><td>CRC 输入反转控制： 0: 输入不反转 1: 输入按字节反转</td></tr><tr><td>CRC 输出反转控制： 0: 输出不反转 1: 输出按字节反转</td></tr><tr><td>输出异或使能： DISABLE: 禁止输出异或 ENABLE: 使能输出异或</td></tr><tr><td>运算结果异或值: 0-FFFF</td></tr></table>	CRC 校验多项式选择： 00: CRC16 01: CRC8 10/11: CCITT	CRC 输入反转控制： 0: 输入不反转 1: 输入按字节反转	CRC 输出反转控制： 0: 输出不反转 1: 输出按字节反转	输出异或使能： DISABLE: 禁止输出异或 ENABLE: 使能输出异或	运算结果异或值: 0-FFFF
CRC 校验多项式选择： 00: CRC16 01: CRC8 10/11: CCITT						
CRC 输入反转控制： 0: 输入不反转 1: 输入按字节反转						
CRC 输出反转控制： 0: 输出不反转 1: 输出按字节反转						
输出异或使能： DISABLE: 禁止输出异或 ENABLE: 使能输出异或						
运算结果异或值: 0-FFFF						
输出参数	无					
返回值	无					

### 15.2.2 CRC\_Deinit

函数名	CRC_Deinit
函数原型	void CRC_Deinit(void)
功能描述	CRC 初始化为默认值
输入参数 1	无
输出参数	无
返回值	无

### 15.2.3 CRC\_CRCDR\_Write

函数名	CRC_CRCDR_Write
-----	-----------------

上海复旦微电子集团股份有限公司

Shanghai Fudan Microelectronics Group Company Limited

FM33A0xx 系列 ARM 固件函数库用户手册

版本 1.0

248

函数原型	void CRC_CRCDDR_Write(uint32_t SetValue)
功能描述	填充 CRC 数据输入
输入参数 1	CRC 数据
输出参数	无
返回值	无

#### 15.2.4 CRC\_CRCDDR\_Read

函数名	CRC_CRCDDR_Read
函数原型	uint32_t CRC_CRCDDR_Read(void)
功能描述	获取 CRC 数据输入
输入参数 1	无
输出参数	无
返回值	CRC 数据

#### 15.2.5 CRC\_CRCCR\_RFLTIN\_Setable

函数名	CRC_CRCCR_RFLTIN_Setable
函数原型	void CRC_CRCCR_RFLTIN_Setable(FunState NewState)
功能描述	配置 CRC 输入反转
输入参数 1	CRC 输入反转： DISABLE：输入不反转 ENABLE：输入按字节反转
输出参数	无
返回值	无

#### 15.2.6 CRC\_CRCCR\_RFLTIN\_Getable

函数名	CRC_CRCCR_RFLTIN_Getable
函数原型	FunState CRC_CRCCR_RFLTIN_Getable(void)
功能描述	获取 CRC 输入反转配置
输入参数 1	无
输出参数	无
返回值	CRC 输入反转： DISABLE：输入不反转 ENABLE：输入按字节反转

**15.2.7 CRC\_CRCCR\_RFLTO\_Setable**

函数名	CRC_CRCCR_RFLTO_Setable
函数原型	void CRC_CRCCR_RFLTO_Setable(FunState NewState)
功能描述	配置 CRC 输出反转
输入参数 1	CRC 输出反转： DISABLE: 输出不反转 ENABLE: 输出按字节反转
输出参数	无
返回值	无

**15.2.8 CRC\_CRCCR\_RFLTO\_Getable**

函数名	CRC_CRCCR_RFLTO_Getable
函数原型	FunState CRC_CRCCR_RFLTO_Getable(void)
功能描述	获取 CRC 输出反转配置
输入参数 1	无
输出参数	无
返回值	CRC 输出反转： DISABLE: 输出不反转 ENABLE: 输出按字节反转

**15.2.9 CRC\_CRCCR\_RES\_Chk**

函数名	CRC_CRCCR_RES_Chk
函数原型	FlagStatus CRC_CRCCR_RES_Chk(void)
功能描述	获取 CRC 结果标志位
输入参数 1	无
输出参数	无
返回值	CRC 结果标志位： RESET: CRC 结果全 0 SET: CRC 结果非全 0

**15.2.10 CRC\_CRCCR\_BUSY\_Chk**

函数名	CRC_CRCCR_BUSY_Chk
函数原型	FlagStatus CRC_CRCCR_BUSY_Chk(void)
功能描述	获取 CRC 运算标志位

输入参数 1	无
输出参数	无
返回值	CRC 运算标志位： RESET: CRC 运算结束 SET: CRC 运算进行中

### 15.2.11 CRC\_CRCCR\_XOR\_Setable

函数名	CRC_CRCCR_XOR_Setable
函数原型	void CRC_CRCCR_XOR_Setable(FunState NewState)
功能描述	配置 CRC 输出异或使能
输入参数 1	CRC 输出异或使能： DISABLE: 禁止输出异或 CRC_XOR ENABLE : 使能输出异或 CRC_XOR
输出参数	无
返回值	无

### 15.2.12 CRC\_CRCCR\_XOR\_Getable

函数名	CRC_CRCCR_XOR_Getable
函数原型	FunState CRC_CRCCR_XOR_Getable(void)
功能描述	获取 CRC 输出异或使能配置
输入参数 1	无
输出参数	无
返回值	CRC 输出异或使能： DISABLE: 禁止输出异或 CRC_XOR ENABLE : 使能输出异或 CRC_XOR

### 15.2.13 CRC\_CRCCR\_CRCSEL\_Set

函数名	CRC_CRCCR_CRCSEL_Set
函数原型	void CRC_CRCCR_CRCSEL_Set(uint32_t SetValue)
功能描述	配置 CRC 校验多项式选择
输入参数 1	CRC 校验多项式选择： CRC_CRCCR_CRCSEL_CRC16: CRC16 CRC_CRCCR_CRCSEL_CRC8: CRC8 CRC_CRCCR_CRCSEL_CCITT: CCITT
输出参数	无

返回值	无
-----	---

#### 15.2.14 CRC\_CRCCR\_CRCSEL\_Get

函数名	CRC_CRCCR_CRCSEL_Get
函数原型	uint32_t CRC_CRCCR_CRCSEL_Get(void)
功能描述	获取 CRC 校验多项式选择
输入参数 1	无
输出参数	无
返回值	CRC 校验多项式选择: CRC_CRCCR_CRCSEL_CRC16: CRC16 CRC_CRCCR_CRCSEL_CRC8: CRC8 CRC_CRCCR_CRCSEL_CCITT: CCITT

#### 15.2.15 CRC\_CRCLFSR\_Write

函数名	CRC_CRCLFSR_Write
函数原型	void CRC_CRCLFSR_Write(uint32_t SetValue)
功能描述	配置 CRC 运算初值
输入参数 1	CRC 运算初值: 0-FFFF
输出参数	无
返回值	无

#### 15.2.16 CRC\_CRCLFSR\_Read

函数名	CRC_CRCLFSR_Read
函数原型	uint32_t CRC_CRCLFSR_Read(void)
功能描述	获取 CRC 运算初值
输入参数 1	无
输出参数	无
返回值	CRC 运算初值: 0-FFFF

#### 15.2.17 CRC\_CRCXOR\_Write

函数名	CRC_CRCXOR_Write
函数原型	void CRC_CRCXOR_Write(uint32_t SetValue)
功能描述	配置 CRC 输出异或值
输入参数 1	CRC 输出异或值: 0-FFFF



输出参数	无
返回值	无

### 15.2.18 CRC\_CRCXOR\_Read

函数名	CRC_CRCXOR_Read
函数原型	uint32_t CRC_CRCXOR_Read(void)
功能描述	获取 CRC 输出异或值
输入参数 1	无
输出参数	无
返回值	CRC 输出异或值：0-FFFF

### 15.2.19 CRC\_CRCFLSEN\_FLSCRCEN\_Setable

函数名	CRC_CRCFLSEN_FLSCRCEN_Setable
函数原型	void CRC_CRCFLSEN_FLSCRCEN_Setable(FunState NewState)
功能描述	配置 CRC 校验使能
输入参数 1	CRC 校验使能： DISABLE：禁止 Flash CRC 校验 ENABLE：使能 Flash CRC 校验
输出参数	无
返回值	无

### 15.2.20 CRC\_CRCFLSEN\_FLSCRCEN\_Getable

函数名	CRC_CRCFLSEN_FLSCRCEN_Getable
函数原型	FunState CRC_CRCFLSEN_FLSCRCEN_Getable(void)
功能描述	获取 CRC 校验使能配置
输入参数 1	无
输出参数	无
返回值	CRC 校验使能： DISABLE：禁止 Flash CRC 校验 ENABLE：使能 Flash CRC 校验

### 15.2.21 CRC\_CRCFLSAD\_Write

函数名	CRC_CRCFLSAD_Write
函数原型	void CRC_CRCFLSAD_Write(uint32_t SetValue)



功能描述	配置 FLASH CRC 校验起始地址
输入参数 1	FLASH CRC 校验起始地址: 0-1FFFF
输出参数	无
返回值	无

### 15.2.22 CRC\_CRCFLSAD\_Read

函数名	CRC_CRCFLSAD_Read
函数原型	uint32_t CRC_CRCFLSAD_Read(void)
功能描述	获取 FLASH CRC 校验起始地址
输入参数 1	无
输出参数	无
返回值	FLASH CRC 校验起始地址: 0-1FFFF

### 15.2.23 CRC\_CRCFLSSIZE\_Write

函数名	CRC_CRCFLSSIZE_Write
函数原型	void CRC_CRCFLSSIZE_Write(uint32_t SetValue)
功能描述	配置 FLASH CRC 校验长度
输入参数 1	FLASH CRC 校验长度 FLSS: 0-1FFFF, 实际运算长度为(FLSS+1) *4 字节
输出参数	无
返回值	无

### 15.2.24 CRC\_CRCFLSSIZE\_Read

函数名	CRC_CRCFLSSIZE_Read
函数原型	uint32_t CRC_CRCFLSSIZE_Read(void)
功能描述	获取 FLASH CRC 校验长度
输入参数 1	无
输出参数	无
返回值	FLASH CRC 校验长度 FLSS: 0-1FFFF, 实际运算长度为(FLSS+1) *4 字节

# 16 DMA

## 16.1 DMA 寄存器结构

寄存器	描述
GLOBALCTRL	DMA 全局控制寄存器
CH0CTRL	通道 0 控制寄存器
CH0RAMAD	通道 0RAM 地址寄存器
CH1CTRL	通道 1 控制寄存器
CH1RAMAD	通道 1RAM 地址寄存器
CH2CTRL	通道 2 控制寄存器
CH2RAMAD	通道 2RAM 地址寄存器
CH3CTRL	通道 3 控制寄存器
CH3RAMAD	通道 3RAM 地址寄存器
CH4CTRL	通道 4 控制寄存器
CH4RAMAD	通道 4RAM 地址寄存器
CH5CTRL	通道 5 控制寄存器
CH5RAMAD	通道 5RAM 地址寄存器
CH6CTRL	通道 6 控制寄存器
CH6RAMAD	通道 6RAM 地址寄存器
CH7CTRL	通道 7 控制寄存器
CH7FLSAD	通道 7Flash 地址寄存器
CH7RAMAD	通道 7RAM 地址寄存器
CHSTATUS	DMA 通道状态标志寄存器

## 16.2 DMA 库函数

序号	函数名	描述
1	DMA_Init	DMA 初始化
2	DMA_Deinit	DMA 初始化为缺省值
3	DMA_GLOBALCTRL_DMAEN_Setable	设置 DMA 功能全局使能
4	DMA_GLOBALCTRL_DMAEN_Getable	获取 DMA 功能全局使能设置
5	DMA_CHxCTRL_EN_Setable	设置 Channelx 启动
6	DMA_CHxCTRL_TSIZE_Set	设置 Channelx 传输长度
7	DMA_CHxCTRL_TSIZE_Get	获取 Channelx 传输长度



8	DMA_CHxCTRL_PRI_Set	设置 Channelx 优先级
9	DMA_CHxCTRL_PRI_Get	获取 Channelx 优先级
10	DMA_CHxCTRL_INC_Set	设置 Channelx RAM 地址增减设置
11	DMA_CHxCTRL_INC_Get	获取 Channelx RAM 地址增减设置
12	DMA_CHxCTRL_SSEL_Set	设置 Channelx 外设通道选择
13	DMA_CHxCTRL_SSEL_Get	获取 Channelx 外设通道选择
14	DMA_CHxCTRL_FTIE_Setable	设置 Channelx 传输中断完成使能
15	DMA_CHxCTRL_FTIE_Getable	获取 Channelx 传输中断完成使能
16	DMA_CHxCTRL_HTIE_Setable	设置 Channelx 半程传输完成中断使能
17	DMA_CHxCTRL_HTIE_Getable	设置 Channelx 半程传输完成中断使能
18	DMA_CHSTATUS_DMACHxHT_Clr	清 Channelx 传输半程标志
19	DMA_CHSTATUS_DMACHxHT_Chk	获取 Channelx 传输半程标志
20	DMA_CHSTATUS_DMACHxFT_Clr	清 Channelx 传输完成标志
21	DMA_CHSTATUS_DMACHxFT_Chk	获取 Channelx 传输完成标志
22	DMA_CHxRAMAD_Write	设置 Channelx RAM 指针地址
23	DMA_CH7FLSAD_Write	设置 Channel7 FLASH 指针地址
24	DMA_CH7RAMAD_Write	设置 Channel7 RAM 指针地址

### 16.2.1 DMA\_Init

函数名	DMA_Init				
函数原型	void DMA_Init(DMA_InitTypeDef* para)				
功能描述	DMA 初始化				
输入参数 1	<div>DMA 参数:</div> <table><tr><td>DMA 通道号: CH0~CH7</td></tr><tr><td>通道传输长度: 0-1FFF</td></tr><tr><td>通道优先级: DMA_CHxCTRL_PRI_LOW: 低优先级 DMA_CHxCTRL_PRI_MEDIUM: 优先级 DMA_CHxCTRL_PRI_HIGH: 高优先级 DMA_CHxCTRL_PRI_VERY_HIGH: 高优先级</td></tr><tr><td>CH0~CH6 通道地址增长方向, CH7 无意义:</td></tr></table>	DMA 通道号: CH0~CH7	通道传输长度: 0-1FFF	通道优先级: DMA_CHxCTRL_PRI_LOW: 低优先级 DMA_CHxCTRL_PRI_MEDIUM: 优先级 DMA_CHxCTRL_PRI_HIGH: 高优先级 DMA_CHxCTRL_PRI_VERY_HIGH: 高优先级	CH0~CH6 通道地址增长方向, CH7 无意义:
DMA 通道号: CH0~CH7					
通道传输长度: 0-1FFF					
通道优先级: DMA_CHxCTRL_PRI_LOW: 低优先级 DMA_CHxCTRL_PRI_MEDIUM: 优先级 DMA_CHxCTRL_PRI_HIGH: 高优先级 DMA_CHxCTRL_PRI_VERY_HIGH: 高优先级					
CH0~CH6 通道地址增长方向, CH7 无意义:					

	0：RAM 地址递减 1：RAM 地址递增							
	CH0~CH6 外设通道选择(DMA_CHx=DMA_CH0~DMA_CH6):							
		CH0	CH2	CH2	CH3	CH4	CH5	CH6
	000	ET4_CAP	ET1_CAP	ET3_CAP	BT34_CAP	ET2_CAP	LPTIM_CAP	LPTIM_CAP
	001	ET2_CAP	AES_OUT	BT12_CAP	ET1_CAP	AES_IN	U78161_RX	U78161_TX
	010	BT34_CAP	I2C_RX	AES_OUT	I2C_TX	I2C_RX	U78160_RX	U78160_TX
	011	ET3_CAP	U78160_RX	I2C_TX	U78161_RX	U78161_TX	UART5_RX	UART5_TX
	100	BT12_CAP	UART3_RX	U78161_TX	UART4_RX	UART4_TX	UART2_RX	UART2_TX
	101	CRC_OUT	UART0_RX	UART3_TX	UART1_RX	UART1_TX	UART1_RX	UART1_TX
	110	AES_IN	SRIO_RX	UART0_TX	UART0_RX	UART0_TX	SPI2_RX	SPI2_RX
	111	ADC_OUT	ADC_OUT	SPI0_TX	SPI1_RX	SPI1_TX	SPI0_RX	SPI0_RX
	CH7 传输方式选择(DMA_CHx=DMA_CH7):							
	CH7 Flash 地址增减设置,仅在 Flash-> RAM 传输中有效: XX0: Flash 地址递增 XX1: Flash 地址递减							
	CH7 RAM 地址增减设置,仅在 Flash-> RAM 传输中有效: XX0: RAM 地址递增 XX1: RAM 地址递减							
	CH7 传输方向: XX0:RAM->Flash 传输 XX1: Flash-> RAM 传输							
	通道传输完成中断使能: DISABLE：禁止通道传输完成中断 ENABLE：使能通道传输完成中断							
	通道传输半程中断使能: DISABLE：禁止通道传输半程中断 ENABLE：使能通道传输半程中断							
	通道使能: DISABLE：禁止通道传输 ENABLE：使能通道传输							
	CH0~CH7 通道 RAM 指针地址							
	通道 FLASH 指针地址，仅针对通道 7 有意义							
	输出参数	无						
返回值	无							

### 16.2.2 DMA\_Deinit

函数名	DMA_Deinit
函数原型	void DMA_Deinit(void)
功能描述	DMA 初始化为默认值
输入参数 1	无



输出参数	无
返回值	无

### 16.2.3 DMA\_GLOBALCTRL\_DMAEN\_Setable

函数名	DMA_GLOBALCTRL_DMAEN_Setable
函数原型	void DMA_GLOBALCTRL_DMAEN_Setable(FunState NewState)
功能描述	配置 DMA 功能全局使能
输入参数 1	DMA 功能全局使能: DISABLE: 禁止 DMA ENABLE: 使能 DMA
输入参数 2	无
输出参数	无
返回值	无

### 16.2.4 DMA\_GLOBALCTRL\_DMAEN\_Getable

函数名	DMA_GLOBALCTRL_DMAEN_Getable
函数原型	FunState DMA_GLOBALCTRL_DMAEN_Getable(void)
功能描述	获取 DMA 功能全局使能配置
输入参数 1	无
输出参数	无
返回值	DMA 功能全局使能: DISABLE: 禁止 DMA ENABLE: 使能 DMA

### 16.2.5 DMA\_CHxCTRL\_EN\_Setable

函数名	DMA_CHxCTRL_EN_Setable
函数原型	void DMA_CHxCTRL_EN_Setable(DMA_CH_Type CHx, FunState NewState)
功能描述	配置 Channelx 使能
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输入参数 2	Channelx 使能: DISABLE: 禁止 CHx DMA 功能 ENABLE: 使能 CHx DMA 功能
输出参数	无
返回值	无

### 16.2.6 DMA\_CHxCTRL\_TSIZE\_Set

函数名	DMA_CHxCTRL_TSIZE_Set
函数原型	void DMA_CHxCTRL_TSIZE_Set(DMA_CH_Type CHx, uint32_t SetValue)
功能描述	配置 Channelx 传输长度
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输入参数 2	CHx 传输长度
输出参数	无
返回值	无

### 16.2.7 DMA\_CHxCTRL\_TSIZE\_Get

函数名	DMA_CHxCTRL_TSIZE_Get
函数原型	uint32_t DMA_CHxCTRL_TSIZE_Get(DMA_CH_Type CHx)
功能描述	获取 Channelx 传输长度
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输出参数	无
返回值	CHx 传输长度

### 16.2.8 DMA\_CHxCTRL\_PRI\_Set

函数名	DMA_CHxCTRL_PRI_Set
函数原型	void DMA_CHxCTRL_PRI_Set(DMA_CH_Type CHx, uint32_t SetValue)
功能描述	配置 Channelx 优先级
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输入参数 2	Channelx 优先级: DMA_CHxCTRL_PRI_LOW: 低优先级 DMA_CHxCTRL_PRI_MEDIUM: 中优先级 DMA_CHxCTRL_PRI_HIGH: 高优先级 DMA_CHxCTRL_PRI_VERY_HIGH: 最高优先级
输出参数	无
返回值	无

### 16.2.9 DMA\_CHxCTRL\_PRI\_Get

函数名	DMA_CHxCTRL_PRI_Get
函数原型	uint32_t DMA_CHxCTRL_PRI_Get(DMA_CH_Type CHx)
功能描述	获取 Channelx 优先级

输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输出参数	无
返回值	Channelx 优先级: DMA_CHxCTRL_PRI_LOW: 低优先级 DMA_CHxCTRL_PRI_MEDIUM: 中优先级 DMA_CHxCTRL_PRI_HIGH: 高优先级 DMA_CHxCTRL_PRI_VERY_HIGH: 最高优先级

### 16.2.10 DMA\_CHxCTRL\_INC\_Set

函数名	DMA_CHxCTRL_INC_Set
函数原型	void DMA_CHxCTRL_INC_Set(DMA_CH_Type CHx, uint32_t SetValue)
功能描述	配置 Channelx RAM 地址增减设置
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输入参数 2	Channelx RAM 地址增减设置: DMA_CHxCTRL_INC_DEC: 地址递减 DMA_CHxCTRL_INC_INC: 地址递增
输出参数	无
返回值	无

### 16.2.11 DMA\_CHxCTRL\_INC\_Get

函数名	DMA_CHxCTRL_INC_Get
函数原型	uint32_t DMA_CHxCTRL_INC_Get(DMA_CH_Type CHx)
功能描述	获取 Channelx RAM 地址增减设置
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输出参数	无
返回值	Channelx RAM 地址增减设置: DMA_CHxCTRL_INC_DEC: 地址递减 DMA_CHxCTRL_INC_INC: 地址递增

### 16.2.12 DMA\_CHxCTRL\_SSEL\_Set

函数名	DMA_CHxCTRL_SSEL_Set
函数原型	void DMA_CHxCTRL_SSEL_Set(DMA_CH_Type CHx, uint32_t SetValue)
功能描述	配置 Channelx 外设通道选择
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输入参数 2	Channelx 外设通道选择:



	DMA_CH0CTRL_CH0SSEL_ET4_CAP: ET4 捕获通道 DMA_CH0CTRL_CH0SSEL_ET2_CAP: ET2 捕获通道 DMA_CH0CTRL_CH0SSEL_BT2_CAP: BT2 捕获通道 DMA_CH0CTRL_CH0SSEL_ET3_CAP: ET3 捕获通道 DMA_CH0CTRL_CH0SSEL_BT1_CAP: BT1 捕获通道 DMA_CH0CTRL_CH0SSEL_CRC_OUT: CRC 输出通道 DMA_CH0CTRL_CH0SSEL_AES_IN: AES 输入通道 DMA_CH0CTRL_CH0SSEL_ADC_OUT: ADC 输出通道
输出参数	无
返回值	无

### 16.2.13 DMA\_CHxCTRL\_SSEL\_Get

函数名	DMA_CHxCTRL_SSEL_Get
函数原型	uint32_t DMA_CHxCTRL_SSEL_Get(DMA_CH_Type CHx)
功能描述	获取 Channelx 外设通道选择
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输出参数	无
返回值	Channelx 外设通道选择: DMA_CH0CTRL_CH0SSEL_ET4_CAP: ET4 捕获通道 DMA_CH0CTRL_CH0SSEL_ET2_CAP: ET2 捕获通道 DMA_CH0CTRL_CH0SSEL_BT2_CAP: BT2 捕获通道 DMA_CH0CTRL_CH0SSEL_ET3_CAP: ET3 捕获通道 DMA_CH0CTRL_CH0SSEL_BT1_CAP: BT1 捕获通道 DMA_CH0CTRL_CH0SSEL_CRC_OUT: CRC 输出通道 DMA_CH0CTRL_CH0SSEL_AES_IN: AES 输入通道 DMA_CH0CTRL_CH0SSEL_ADC_OUT: ADC 输出通道

### 16.2.14 DMA\_CHxCTRL\_FTIE\_Setable

函数名	DMA_CHxCTRL_FTIE_Setable
函数原型	void DMA_CHxCTRL_FTIE_Setable(DMA_CH_Type CHx, FunState NewState)
功能描述	配置 Channelx 传输中断完成使能
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输入参数 2	Channelx 传输中断完成使能: DISABLE: 禁止传输完成中断 ENABLE: 使能传输完成中断
输出参数	无



返回值	无
-----	---

### 16.2.15 DMA\_CHxCTRL\_FTIE\_Getable

函数名	DMA_CHxCTRL_FTIE_Getable
函数原型	FunState DMA_CHxCTRL_FTIE_Getable(DMA_CH_Type CHx)
功能描述	获取 Channelx 传输中断完成使能
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输出参数	无
返回值	Channelx 传输中断完成使能: DISABLE: 禁止传输完成中断 ENABLE: 使能传输完成中断

### 16.2.16 DMA\_CHxCTRL\_HTIE\_Setable

函数名	DMA_CHxCTRL_HTIE_Setable
函数原型	void DMA_CHxCTRL_HTIE_Setable(DMA_CH_Type CHx, FunState NewState)
功能描述	配置 Channelx 半程传输完成中断使能
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输入参数 2	Channelx 半程传输完成中断完成使能: DISABLE: 禁止半程传输完成中断 ENABLE: 使能半程传输完成中断
输出参数	无
返回值	无

### 16.2.17 DMA\_CHxCTRL\_HTIE\_Getable

函数名	DMA_CHxCTRL_HTIE_Getable
函数原型	FunState DMA_CHxCTRL_HTIE_Getable(DMA_CH_Type CHx)
功能描述	配置 Channelx 半程传输完成中断使能
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输出参数	无
返回值	Channelx 半程传输完成中断完成使能: DISABLE: 禁止半程传输完成中断 ENABLE: 使能半程传输完成中断

**16.2.18 DMA\_CHSTATUS\_DMACHxHT\_Clr**

函数名	DMA_CHSTATUS_DMACHxHT_Clr
函数原型	void DMA_CHSTATUS_DMACHxHT_Clr(DMA_CH_Type CHx)
功能描述	清 Channelx 传输半程标志
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输出参数	无
返回值	无

**16.2.19 DMA\_CHSTATUS\_DMACHxHT\_Chk**

函数名	DMA_CHSTATUS_DMACHxHT_Chk
函数原型	FlagStatus DMA_CHSTATUS_DMACHxHT_Chk(DMA_CH_Type CHx)
功能描述	获取 Channelx 传输半程标志
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输出参数	无
返回值	Channelx 传输半程标志: RESET: 对应通道半程传输未完成 SET: 对应通道半程传输完成

**16.2.20 DMA\_CHSTATUS\_DMACHxFT\_Clr**

函数名	DMA_CHSTATUS_DMACHxFT_Clr
函数原型	void DMA_CHSTATUS_DMACHxFT_Clr(DMA_CH_Type CHx)
功能描述	清 Channelx 传输完成标志
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输出参数	无
返回值	无

**16.2.21 DMA\_CHSTATUS\_DMACHxFT\_Chk**

函数名	DMA_CHSTATUS_DMACHxFT_Chk
函数原型	FlagStatus DMA_CHSTATUS_DMACHxFT_Chk(DMA_CH_Type CHx)
功能描述	获取 Channelx 传输完成标志
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输出参数	无
返回值	Channelx 传输完成标志: RESET: 对应通道传输未完成

SET: 对应通道传输完成

**16.2.22 DMA\_CHxRAMAD\_Write**

函数名	DMA_CHxRAMAD_Write
函数原型	void DMA_CHxRAMAD_Write(DMA_CH_Type CHx, uint32_t SetValue)
功能描述	配置 Channelx RAM 指针地址(DMA 传输启动前软件向此寄存器写入 RAM 目标地址, DMA 启动后此寄存器随 DMA 传输自增或自减)
输入参数 1	DMA 通道: DMA_CH0~ DMA_CH7
输入参数 2	16bit 地址
输出参数	无
返回值	无

**16.2.23 DMA\_CH7FLSAD\_Write**

函数名	DMA_CH7FLSAD_Write
函数原型	void DMA_CH7FLSAD_Write(uint32_t SetValue)
功能描述	配置 Channel7 FLASH 指针地址(DMA 传输启动前软件向此寄存器写入 Flash 目标地址, DMA 启动后此寄存器随 DMA 传输自增或自减)
输入参数 1	17bit 地址
输出参数	无
返回值	无

**16.2.24 DMA\_CH7RAMAD\_Write**

函数名	DMA_CH7RAMAD_Write
函数原型	void DMA_CH7RAMAD_Write(uint32_t SetValue)
功能描述	配置 Channel7 RAM 指针地址(DMA 传输启动前软件向此寄存器写入 RAM 目标地址, DMA 启动后此寄存器随 DMA 传输自增或自减)
输入参数 1	14bit 地址
输出参数	无
返回值	无

# 17 TRNG

## 17.1 TRNG 寄存器结构

寄存器	描述
TRNGCON	TRNG 控制寄存器
RNGOUT	随机数结果输出寄存器
RNGIE	RNG 中断使能寄存器
RNGIF	RNG 中断标志寄存器
CRCCON	CRC 控制寄存器
CRCIN	CRC 输入数据寄存器
CRCFLAG	CRC 标志寄存器

## 17.2 TRNG 库函数

序号	函数名	描述
1	TRNG_Deinit	RNG 初始化为缺省值
2	TRNG_RNGOUT_Read	获取随机数输出
3	TRNG_CRCCON_CRCEN_Setable	设置随机数 CRC 使能
4	TRNG_CRCCON_CRCEN_Getable	获取随机数 CRC 使能
5	TRNG_RNGIE_RNGIE_Setable	设置随机数生成失败中断使能
6	TRNG_RNGIE_RNGIE_Getable	获取随机数生成失败中断使能
7	TRNG_RNGIF_LFSREN_Chk	获取随机数 LFSR 使能标志
8	TRNG_RNGIF_RNF_Clr	清随机数 LFSR 使能标志
9	TRNG_RNGIF_RNF_Chk	获取随机数生成失败标志
10	TRNG_CRCIN_Write	设置随机数 CRC 运算数据输入
11	TRNG_CRCIN_Read	获取随机数 CRC 运算数据输入
12	TRNG_CRCFLAG_CRCD_Clr	清 CRC 计算完成标志
13	TRNG_CRCFLAG_CRCD_Chk	获取 CRC 计算完成标志
14	ANAC_TRNGCON_TRNGEN_Setable	设置 TRNG 使能
15	ANAC_TRNGCON_TRNGEN_Getable	获取 TRNG 使能状态

### 17.2.1 TRNG\_Deinit

函数名	TRNG_Deinit
-----	-------------

函数原型	void TRNG_Deinit(void)
功能描述	RNG 初始化为默认值
输入参数 1	无
输出参数	无
返回值	无

### 17.2.2 TRNG\_RNGOUT\_Read

函数名	TRNG_RNGOUT_Read
函数原型	uint32_t TRNG_RNGOUT_Read(void)
功能描述	获取随机数输出
输入参数 1	无
输出参数	无
返回值	随机数生成结果

### 17.2.3 TRNG\_CRCCON\_CRCEN\_Setable

函数名	TRNG_CRCCON_CRCEN_Setable
函数原型	void TRNG_CRCCON_CRCEN_Setable(FunState NewState)
功能描述	配置随机数 CRC 使能
输入参数 1	随机数 CRC 使能： DISABLE：禁止随机数 CRC ENABLE：使能随机数 CRC
输出参数	无
返回值	无

### 17.2.4 TRNG\_CRCCON\_CRCEN\_Getable

函数名	TRNG_CRCCON_CRCEN_Getable
函数原型	FunState TRNG_CRCCON_CRCEN_Getable(void)
功能描述	获取随机数 CRC 使能
输入参数 1	无
输出参数	无
返回值	随机数 CRC 使能： DISABLE：禁止随机数 CRC ENABLE：使能随机数 CRC

### 17.2.5 TRNG\_RNGIE\_RNGIE\_Setable

函数名	TRNG_RNGIE_RNGIE_Setable
函数原型	void TRNG_RNGIE_RNGIE_Setable(FunState NewState)
功能描述	配置随机数生成失败中断使能
输入参数 1	随机数生成失败中断使能： DISABLE：禁止随机数生成失败中断 ENABLE：使能随机数生成失败中断
输出参数	无
返回值	无

### 17.2.6 TRNG\_RNGIE\_RNGIE\_Getable

函数名	TRNG_RNGIE_RNGIE_Getable
函数原型	FunState TRNG_RNGIE_RNGIE_Getable(void)
功能描述	获取随机数生成失败中断使能
输入参数 1	无
输出参数	无
返回值	随机数生成失败中断使能： DISABLE：禁止随机数生成失败中断 ENABLE：使能随机数生成失败中断

### 17.2.7 TRNG\_RNGIF\_LFSREN\_Chk

函数名	TRNG_RNGIF_LFSREN_Chk
函数原型	FlagStatus TRNG_RNGIF_LFSREN_Chk(void)
功能描述	获取随机数 LFSR 使能标志
输入参数 1	无
输出参数	无
返回值	随机数 LFSR 使能标志： RESET：LFSR 不在运行中，可进行 CRC 验证 SET：LFSR 在运行中，不可进行 CRC 验证

### 17.2.8 TRNG\_RNGIF\_RNF\_Clr

函数名	TRNG_RNGIF_RNF_Clr
函数原型	void TRNG_RNGIF_RNF_Clr(void)
功能描述	清随机数 LFSR 使能标志

输入参数 1	无
输出参数	无
返回值	无

### 17.2.9 TRNG\_RNGIF\_RNF\_Chk

函数名	TRNG_RNGIF_RNF_Chk
函数原型	FlagStatus TRNG_RNGIF_RNF_Chk(void)
功能描述	获取随机数生成失败标志
输入参数 1	无
输出参数	无
返回值	随机数生成失败标志： RESET：随机数通过质量检测 SET：随机数未能通过质量检测

### 17.2.10 TRNG\_CRCIN\_Write

函数名	TRNG_CRCIN_Write
函数原型	void TRNG_CRCIN_Write(uint32_t SetValue)
功能描述	配置随机数 CRC 运算数据输入
输入参数 1	随机数 CRC 运算输入数据:0-FFFFFFFF
输出参数	无
返回值	无

### 17.2.11 TRNG\_CRCIN\_Read

函数名	TRNG_CRCIN_Read
函数原型	uint32_t TRNG_CRCIN_Read(void)
功能描述	获取随机数 CRC 运算数据输入
输入参数 1	无
输出参数	无
返回值	随机数 CRC 运算输入数据:0-FFFFFFFF

### 17.2.12 TRNG\_CRCFLAG\_CRCD\_Clr

函数名	TRNG_CRCFLAG_CRCD_Clr
函数原型	void TRNG_CRCFLAG_CRCD_Clr(void)
功能描述	清 CRC 计算完成标志

输入参数 1	无
输出参数	无
返回值	无

### 17.2.13 TRNG\_CRCFLAG\_CRCD\_Chk

函数名	TRNG_CRCFLAG_CRCD_Chk
函数原型	FlagStatus TRNG_CRCFLAG_CRCD_Chk(void)
功能描述	获取 CRC 计算完成标志
输入参数 1	无
输出参数	无
返回值	CRC 计算完成标志： RESET：未完成 CRC 计算 SET：完成 CRC 计算

### 17.2.14 ANAC\_TRNGCON\_TRNGEN\_Setable

函数名	ANAC_TRNGCON_TRNGEN_Setable
函数原型	void ANAC_TRNGCON_TRNGEN_Setable(FunState NewState)
功能描述	设置 TRNG 使能
输入参数 1	TRNG 使能： DISABLE：禁止 TRNG ENABLE：使能 TRNG
输出参数	无
返回值	无

### 17.2.15 ANAC\_TRNGCON\_TRNGEN\_Getable

函数名	ANAC_TRNGCON_TRNGEN_Getable
函数原型	FunState ANAC_TRNGCON_TRNGEN_Getable(void)
功能描述	获取 TRNG 使能状态
输入参数 1	无
输出参数	无
返回值	TRNG 使能： DISABLE：禁止 TRNG ENABLE：使能 TRNG



# 18 U7816

## 18.1 U7816 寄存器结构

寄存器	描述
U7816CTRL0	U7816 通道控制寄存器
U7816CTRL1	
U7816FRC0	U7816 帧格式控制寄存器
U7816FRC1	
U7816EGT0	U7816EGT 设置寄存器
U7816EGT1	
U7816CLKDIV0	U7816 工作时钟分频寄存器
U7816CLKDIV1	
U7816PDIV0	U7816 预分频寄存器
U7816PDIV1	
U7816RXBUF0	U7816 数据接收缓存寄存器
U7816RXBUF1	
U7816TXBUF0	U7816 数据发送缓存寄存器
U7816TXBUF1	
U7816IE0	U7816 中断使能寄存器
U7816IE1	
U7816IF0	U7816 主状态中断标志寄存器
U7816IF1	
U7816ERR0	U7816 错误标志寄存器
U7816ERR1	
U7816STA0	U7816 次状态寄存器
U7816STA1	

## 18.2 U7816 库函数

序号	函数名	描述
1	U7816_Init	U7816 初始化
2	U7816x_Deinit	U7816 初始化为缺省值
3	U7816x_U7816CTRL_TXEN_Setable	U7816 通道发送使能控
4	U7816x_U7816CTRL_TXEN_Getable	获取 U7816 通道发送使能控制状



		态
5	U7816x_U7816CTRL_RXEN_Setable	设置 7816 通道接收使能控制
6	U7816x_U7816CTRL_RXEN_Getable	获取 U7816 通道接收使能控制状态
7	U7816CTRL_CKOEN_Setable	设置 U7816 时钟 CLK 输出使能控制
8	U7816CTRL_CKOEN_Getable	获取 U7816 时钟 CLK 输出使能控制状态
9	U7816x_U7816CTRL_HPUAT_Setable	设置 7816 通道数据发送强上拉电阻自动有效控制
10	U7816x_U7816CTRL_HPUAT_Getable	获取 7816 通道数据发送强上拉电阻自动有效控制
11	U7816x_U7816CTRL_HPUEN_Setable	设置 U7816 通道强上拉使能控制
12	U7816x_U7816CTRL_HPUEN_Getable	获取 U7816 通道强上拉使能控制状态
13	U7816x_U7816FRC_ERSW_Set	设置 ERROR SIGNAL 宽度选择
14	U7816x_U7816FRC_ERSW_Get	获取 ERROR SIGNAL 宽度选择
15	U7816x_U7816FRC_ERSGD_Set	设置 ERROR SIGNAL 后 GUARDTIME 宽度选择
16	U7816x_U7816FRC_ERSGD_Get	获取 ERROR SIGNAL 后 GUARDTIME 宽度选择
17	U7816x_U7816FRC_BG TEN_Setable	设置 BGT 控制
18	U7816x_U7816FRC_BG TEN_Getable	获取 BGT 控制
19	U7816x_U7816FRC_REP_T_Set	设置控制接收数据奇偶校验出错时自动重发次数
20	U7816x_U7816FRC_REP_T_Get	获取接收数据奇偶校验出错时自动重发次数
21	U7816x_U7816FRC_PAR_Set	设置奇偶校验类型选择
22	U7816x_U7816FRC_PAR_Get	获取奇偶校验类型选择
23	U7816x_U7816FRC_FREN_Setable	设置 Guard Time 长度控制位
24	U7816x_U7816FRC_FREN_Getable	获取 Guard Time 长度控制位
25	U7816x_U7816FRC_TREPEN_Setable	设置发送数据奇偶校验错的处理方式选择
26	U7816x_U7816FRC_TREPEN_Getable	获取发送数据奇偶校验错的处理方式选择
27	U7816x_U7816FRC_RREPEN_Setable	设置接收数据奇偶校验错的处理方式选择

28	U7816x_U7816FRC_RREPEN_Getable	获取接收数据奇偶校验错误的处理方式选择
29	U7816x_U7816FRC_DICONV_Setable	设置传输次序反向编码使能
30	U7816x_U7816FRC_DICONV_Getable	获取传输次序反向编码状态
31	U7816x_U7816EGT_Write	设置发送时插入的 EGT 时间
32	U7816x_U7816EGT_Read	: 获取发送时插入的 EGT 时间
33	U7816x_U7816CLKDIV_Write	设置 U7816 时钟输出分频控制
34	U7816x_U7816CLKDIV_Read	获取 U7816 时钟输出分频控制值
35	U7816x_U7816PDIV_Write	设置 U7816 预分频控制, 控制 7816 通信分频比
36	U7816x_U7816PDIV_Read	获取 U7816 预分频控制值
37	U7816x_U7816TXBUF_Write	U7816 发送数据
38	U7816x_U7816RXBUF_Read	读 U7816 数据接收缓存寄存器
39	U7816x_U7816IE_RXIE_Setable	数据接收中断使能位
40	U7816x_U7816IE_RXIE_Getable	获取数据接收中断使能位状态
41	U7816x_U7816IE_TXIE_Setable	设置数据发送中断使能
42	U7816x_U7816IE_TXIE_Getable	获取数据发送中断使能状态
43	U7816x_U7816IE_LSIE_Setable	设置线路状态中断使能
44	U7816x_U7816IE_LSIE_Getable	: 获取线路状态中断使能状态
45	U7816x_U7816IF_ERRIF_Chk	获取错误中断标志
46	U7816x_U7816IF_TXIF_Chk	获取发送缓冲区空标志
47	U7816x_U7816IF_RXIF_Chk	获取接收完成标志
48	U7816x_U7816ERR_TPARERR_Clr	清发送数据奇偶校验错误标志
49	U7816x_U7816ERR_TPARERR_Chk	获取发送数据奇偶校验错误标志
50	U7816x_U7816ERR_RPARERR_Clr	清接收数据奇偶校验错误标志
51	U7816x_U7816ERR_RPARERR_Chk	获取接收数据奇偶校验错误标志
52	U7816x_U7816ERR_FRERR_Clr	清收帧格式错误标志位
53	U7816x_U7816ERR_FRERR_Chk	获取接收帧格式错误标志位
54	U7816x_U7816ERR_OVERR_Clr	清接收溢出错误标志位
55	U7816x_U7816ERR_OVERR_Chk	获取收溢出错误标志位
56	U7816x_U7816STA_WAIT_RPT_Chk	获取错误等待标志
57	U7816x_U7816STA_TXBUSY_Chk	获取发送数据忙标志
58	U7816x_U7816STA_RXBUSY_Chk	获取接收数据忙标志

### 18.2.1 U7816\_Init

函数名	U7816_Init
-----	------------



函数原型	void U7816_Init(U7816x_Type* U7816x, U7816_InitTypeDef* para)
功能描述	U7816 初始化
输入参数 1	U78160/U78161
输入参数 2	<div>U7816 参数:</div> <div><div>U7816 发送使能: DISABLE: 禁止 U7816 发送 ENABLE: 使能 U7816 发送</div><div>U7816 接收使能: DISABLE: 禁止 U7816 接收 ENABLE: 使能 U7816 接收</div><div>U7816 时钟 CLK 输出使能控制: DISABLE: 禁止 U7816 时钟 CLK 输出 ENABLE: 使能 U7816 时钟 CLK 输出</div><div>U7816 通道数据发送强上拉电阻自动有效控制: DISABLE: 禁止数据发送时上拉电阻自动有效 ENABLE: 使能数据发送时上拉电阻自动有效</div><div>U7816 通道强上拉使能控制: DISABLE: 禁止 U7816 通道强上拉 ENABLE: 使能 U7816 通道强上拉</div><div>ERROR SIGNAL 宽度选择: 00: ERROR SIGNAL 宽度为 2ETU 01: ERROR SIGNAL 宽度为 2ETU 10: ERROR SIGNAL 宽度为 1.5ETU 11: ERROR SIGNAL 宽度为 1ETU</div><div>ERROR SIGNAL 后 GUARDTIME 宽度选择: 0: ERROR SIGNAL 后 GUARDTIME 宽度为 2~2.5ETU 1: ERROR SIGNAL 后 GUARDTIME 宽度为 1~1.5ETU</div><div>BGT (block guard time) 使能控制: DISABLE: 禁止 BGT ENABLE: 使能 BGT</div><div>控制接收数据奇偶校验出错时自动重发次数: 0: 1 次 1: 3 次</div><div>奇偶校验类型选择: 00: Even 01: Odd 10: Always1 11: 不校验</div><div>Guard Time 长度控制位: 0: Guard Time 为 2ETU 1: Guard Time 为 1ETU</div><div>缓发送数据奇偶校验错的处理方式选择: 0: 收到 ERROR SIGNAL 时不进行自动回发, 置 tx parity err 标志, 直接中断</div></div>

	<p>1: 收到 ERROR SIGNAL 时根据 T=0 协议进行自动回发。在单一 byte 重复发送次数超过 REP_T 后, 置 rx_parity_err 标志, 进行中断</p> <p>接收数据奇偶校验错误的处理方式选择:</p> <p>0: 收到 ERROR SIGNAL 时不进行自动回发, 置 tx_parity_err 标志, 进行中断</p> <p>1: 收到 ERROR SIGNAL 时根据 T=0 协议进行自动回发。在单一 byte 重复发送次数超过 REP_T 后, 置 rx_parity_err 标志, 进行中断</p> <p>传输次序反向编码使能:</p> <p>DISABLE: 禁止传输次序反向编码</p> <p>ENABLE: 使能传输次序反向编码</p> <p>发送时插入的 EGT 时间 (以 ETU 为单位) :0-FF</p> <p>U7816 时钟输出分频数: 0-1F</p> <p>U7816 预分频控制, 控制 7816 通信分频比 (波特率) : 0-FFF</p> <p>数据接收中断使能:</p> <p>DISABLE: 禁止数据接收中断</p> <p>ENABLE: 使能数据接收中断</p> <p>数据发送中断使能:</p> <p>DISABLE: 禁止数据发送中断</p> <p>ENABLE: 使能数据发送中断</p> <p>线路状态中断使能:</p> <p>DISABLE: 禁止线路状态中断</p> <p>ENABLE: 使能线路状态中断</p>	
输出参数	无	
返回值	无	

### 18.2.2 U7816x\_Deinit

函数名	U7816x_Deinit
函数原型	void U7816x_Deinit(U7816x_Type* U7816x)
功能描述	U7816 初始化为默认值
输入参数 1	U78160/U78161
输入参数 2	无
输出参数	无
返回值	无

### 18.2.3 U7816x\_U7816CTRL\_TXEN\_Setable

函数名	U7816x_U7816CTRL_TXEN_Setable
函数原型	void U7816x_U7816CTRL_TXEN_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	U7816 通道发送使能

输入参数 1	U78160/U78161
输入参数 2	U7816 通道发送使能: DISABLE: 禁止 U7816 通道发送 ENABLE: 使能 U7816 通道发送
输出参数	无
返回值	无

#### 18.2.4 U7816x\_U7816CTRL\_TXEN\_Getable

函数名	U7816x_U7816CTRL_TXEN_Getable
函数原型	FunState U7816x_U7816CTRL_TXEN_Getable(U7816x_Type* U7816x)
功能描述	获取 U7816 通道发送使能控制状态
输入参数 1	U78160/U78161
输出参数	无
返回值	U7816 通道发送使能: DISABLE: 禁止 U7816 通道发送 ENABLE: 使能 U7816 通道发送

#### 18.2.5 U7816x\_U7816CTRL\_RXEN\_Setable

函数名	U7816x_U7816CTRL_RXEN_Setable
函数原型	void U7816x_U7816CTRL_RXEN_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	设置 7816 通道接收使能控制
输入参数 1	U78160/U78161
输入参数 2	U7816 通道接收使能控制: DISABLE: 禁止 U7816 通道接收 ENABLE: 使能 U7816 通道接收
输出参数	无
返回值	无

#### 18.2.6 U7816x\_U7816CTRL\_RXEN\_Getable

函数名	U7816x_U7816CTRL_RXEN_Getable
函数原型	FunState U7816x_U7816CTRL_RXEN_Getable(U7816x_Type* U7816x)
功能描述	获取 U7816 通道接收使能控制状态
输入参数 1	U78160/U78161
输出参数	无



返回值	U7816 通道接收使能控制： DISABLE：禁止 U7816 通道接收 ENABLE：使能 U7816 通道接收
-----	------------------------------------------------------------------

### 18.2.7 U7816CTRL\_CKOEN\_Setable

函数名	U7816CTRL_CKOEN_Setable
函数原型	void U7816x_U7816CTRL_CKOEN_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	设置 U7816 时钟 CLK 输出使能控制
输入参数 1	U78160/U78161
输入参数 2	U7816 时钟 CLK 输出使能控制： DISABLE：禁止 U7816 时钟 CLK 输出 ENABLE：使能 U7816 时钟 CLK 输出
输出参数	无
返回值	无

### 18.2.8 U7816CTRL\_CKOEN\_Getable

函数名	U7816CTRL_CKOEN_Getable
函数原型	U7816x_U7816CTRL_CKOEN_Getable
功能描述	获取 U7816 时钟 CLK 输出使能控制状态
输入参数 1	U78160/U78161
输出参数	无
返回值	U7816 时钟 CLK 输出使能控制： DISABLE：禁止 U7816 时钟 CLK 输出 ENABLE：使能 U7816 时钟 CLK 输出

### 18.2.9 U7816x\_U7816CTRL\_HPUAT\_Setable

函数名	U7816x_U7816CTRL_HPUAT_Setable
函数原型	void U7816x_U7816CTRL_HPUAT_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	设置 7816 通道数据发送强上拉电阻自动有效控制
输入参数 1	U78160/U78161
输入参数 2	7816 通道数据发送强上拉电阻自动有效控制： DISABLE：禁止 7816 通道数据发送强上拉电阻自动有效 ENABLE：使能 7816 通道数据发送强上拉电阻自动有效

输出参数	无
返回值	无

### 18.2.10 U7816x\_U7816CTRL\_HPUAT\_Getable

函数名	U7816x_U7816CTRL_HPUAT_Getable
函数原型	FunState U7816x_U7816CTRL_HPUAT_Getable(U7816x_Type* U7816x)
功能描述	获取 7816 通道数据发送强上拉电阻自动有效控制
输入参数 1	U78160/U78161
输出参数	无
返回值	7816 通道数据发送强上拉电阻自动有效控制： DISABLE：禁止 7816 通道数据发送强上拉电阻自动有效 ENABLE：使能 7816 通道数据发送强上拉电阻自动有效

### 18.2.11 U7816x\_U7816CTRL\_HPUEN\_Setable

函数名	U7816x_U7816CTRL_HPUEN_Setable
函数原型	void U7816x_U7816CTRL_HPUEN_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	设置 U7816 通道强上拉使能控制
输入参数 1	U78160/U78161
输入参数 2	U7816 通道强上拉使能控制： DISABLE：禁止 U7816 通道强上拉 ENABLE：使能 U7816 通道强上拉
输出参数	无
返回值	DISABLE/ENABLE

### 18.2.12 U7816x\_U7816CTRL\_HPUEN\_Getable

函数名	U7816x_U7816CTRL_HPUEN_Getable
函数原型	FunState U7816x_U7816CTRL_HPUEN_Getable(U7816x_Type* U7816x)
功能描述	获取 U7816 通道强上拉使能控制状态
输入参数 1	U78160/U78161
输出参数	无
返回值	U7816 通道强上拉使能控制： DISABLE：禁止 U7816 通道强上拉 ENABLE：使能 U7816 通道强上拉



**18.2.13 U7816x\_U7816FRC\_ERSW\_Set**

函数名	U7816x_U7816FRC_ERSW_Set
函数原型	void U7816x_U7816FRC_ERSW_Set(U7816x_Type* U7816x, uint32_t SetValue)
功能描述	设置 ERROR SIGNAL 宽度选择
输入参数 1	U78160/U78161
输入参数 2	ERROR SIGNAL 宽度选择: 00: ERROR SIGNAL 宽度为 2ETU 01: ERROR SIGNAL 宽度为 2ETU 10: ERROR SIGNAL 宽度为 1.5ETU 11: ERROR SIGNAL 宽度为 1ETU
输出参数	无
返回值	无

**18.2.14 U7816x\_U7816FRC\_ERSW\_Get**

函数名	U7816x_U7816FRC_ERSW_Get
函数原型	uint32_t U7816x_U7816FRC_ERSW_Get(U7816x_Type* U7816x)
功能描述	获取 ERROR SIGNAL 宽度选择
输入参数 1	U78160/U78161
输出参数	无
返回值	ERROR SIGNAL 宽度选择: 00: ERROR SIGNAL 宽度为 2ETU 01: ERROR SIGNAL 宽度为 2ETU 10: ERROR SIGNAL 宽度为 1.5ETU 11: ERROR SIGNAL 宽度为 1ETU

**18.2.15 U7816x\_U7816FRC\_ERSGD\_Set**

函数名	U7816x_U7816FRC_ERSGD_Set
函数原型	void U7816x_U7816FRC_ERSGD_Set(U7816x_Type* U7816x, uint32_t SetValue)
功能描述	设置 ERROR SIGNAL 后 GUARDTIME 宽度选择
输入参数 1	U78160/U78161
输入参数 2	ERROR SIGNAL 后 GUARDTIME 宽度选择: 0: ERROR SIGNAL 后 GUARDTIME 宽度为 2~2.5ETU 1: ERROR SIGNAL 后 GUARDTIME 宽度为 1~1.5ETU
输出参数	无
返回值	无

**18.2.16 U7816x\_U7816FRC\_ERSGD\_Get**

函数名	U7816x_U7816FRC_ERSGD_Get
函数原型	uint32_t U7816x_U7816FRC_ERSGD_Get(U7816x_Type* U7816x)
功能描述	获取 ERROR SIGNAL 后 GUARDTIME 宽度选择
输入参数 1	U78160/U78161
输出参数	无
返回值	ERROR SIGNAL 后 GUARDTIME 宽度选择: 0: ERROR SIGNAL 后 GUARDTIME 宽度为 2~2.5ETU 1: ERROR SIGNAL 后 GUARDTIME 宽度为 1~1.5ETU

**18.2.17 U7816x\_U7816FRC\_BG TEN\_Setable**

函数名	U7816x_U7816FRC_BG TEN_Setable
函数原型	void U7816x_U7816FRC_BG TEN_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	设置 BGT 控制
输入参数 1	U78160/U78161
输入参数 2	BGT 使能控制: DISABLE: 禁止 BGT ENABLE: 使能 BGT
输出参数	无
返回值	无

**18.2.18 U7816x\_U7816FRC\_BG TEN\_Getable**

函数名	U7816x_U7816FRC_BG TEN_Getable
函数原型	FunState U7816x_U7816FRC_BG TEN_Getable(U7816x_Type* U7816x)
功能描述	获取 BGT 控制
输入参数 1	U78160/U78161
输出参数	无
返回值	BGT 使能控制: DISABLE: 禁止 BGT ENABLE: 使能 BGT

**18.2.19 U7816x\_U7816FRC\_REP\_T\_Set**

函数名	U7816x_U7816FRC_REP_T_Set
函数原型	void U7816x_U7816FRC_REP_T_Set(U7816x_Type* U7816x, uint32_t SetValue)
功能描述	设置控制接收数据奇偶校验出错时自动重发次数
输入参数 1	U78160/U78161
输入参数 2	控制接收数据奇偶校验出错时自动重发次数： 0: 1 次 1: 3 次
输出参数	无
返回值	DISABLE/ENABLE

**18.2.20 U7816x\_U7816FRC\_REP\_T\_Get**

函数名	U7816x_U7816FRC_REP_T_Get
函数原型	uint32_t U7816x_U7816FRC_REP_T_Get(U7816x_Type* U7816x)
功能描述	获取接收数据奇偶校验出错时自动重发次数
输入参数 1	U78160/U78161
输入参数 2	无
输出参数	无
返回值	控制接收数据奇偶校验出错时自动重发次数： 0: 1 次 1: 3 次

**18.2.21 U7816x\_U7816FRC\_PAR\_Set**

函数名	U7816x_U7816FRC_PAR_Set
函数原型	void U7816x_U7816FRC_PAR_Set(U7816x_Type* U7816x, uint32_t SetValue)
功能描述	设置奇偶校验类型选择
输入参数 1	U78160/U78161
输入参数 2	奇偶校验类型选择： 00: Even 01: Odd 10: Always1 11: 不校验
输出参数	无
返回值	无

**18.2.22 U7816x\_U7816FRC\_PAR\_Get**

函数名	U7816x_U7816FRC_PAR_Get
函数原型	uint32_t U7816x_U7816FRC_PAR_Get(U7816x_Type* U7816x)
功能描述	获取奇偶校验类型选择
输入参数 1	U78160/U78161
输出参数	无
返回值	奇偶校验类型选择： 00: Even 01: Odd 10: Always1 11: 不校验

**18.2.23 U7816x\_U7816FRC\_FREN\_Setable**

函数名	U7816x_U7816FRC_FREN_Setable
函数原型	void U7816x_U7816FRC_FREN_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	设置 Guard Time 长度控制位
输入参数 1	U78160/U78161
输入参数 2	Guard Time 长度控制位： 0: Guard Time 为 2ETU 1: Guard Time 为 1ETU
输出参数	无
返回值	无

**18.2.24 U7816x\_U7816FRC\_FREN\_Getable**

函数名	U7816x_U7816FRC_FREN_Getable
函数原型	FunState U7816x_U7816FRC_FREN_Getable(U7816x_Type* U7816x)
功能描述	获取 Guard Time 长度控制位
输入参数 1	U78160/U78161
输出参数	无
返回值	Guard Time 长度控制位： 0: Guard Time 为 2ETU 1: Guard Time 为 1ETU

**18.2.25 U7816x\_U7816FRC\_TREPEN\_Setable**

函数名	U7816x_U7816FRC_TREPEN_Setable
函数原型	void U7816x_U7816FRC_TREPEN_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	设置发送数据奇偶校验错的处理方式选择
输入参数 1	U78160/U78161
输入参数 2	缓发送数据奇偶校验错的处理方式选择： 0：收到 ERROR SIGNAL 时不进行自动回发，置 tx_parity_err 标志，直接中断 1：收到 ERROR SIGNAL 时根据 T=0 协议进行自动回发。在单一 byte 重复发送次数超过 REP_T 后，置 rx_parity_err 标志，进行中断
输出参数	无
返回值	无

**18.2.26 U7816x\_U7816FRC\_TREPEN\_Getable**

函数名	U7816x_U7816FRC_TREPEN_Getable
函数原型	FunState U7816x_U7816FRC_TREPEN_Getable(U7816x_Type* U7816x)
功能描述	获取发送数据奇偶校验错的处理方式选择
输入参数 1	U78160/U78161
输出参数	无
返回值	缓发送数据奇偶校验错的处理方式选择： 0：收到 ERROR SIGNAL 时不进行自动回发，置 tx_parity_err 标志，直接中断 1：收到 ERROR SIGNAL 时根据 T=0 协议进行自动回发。在单一 byte 重复发送次数超过 REP_T 后，置 rx_parity_err 标志，进行中断

**18.2.27 U7816x\_U7816FRC\_RREPEN\_Setable**

函数名	U7816x_U7816FRC_RREPEN_Setable
函数原型	void U7816x_U7816FRC_RREPEN_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	设置接收数据奇偶校验错的处理方式选择
输入参数 1	U78160/U78161
输入参数 2	接收数据奇偶校验错的处理方式选择： 0：收到 ERROR SIGNAL 时不进行自动回发，置 tx_parity_err 标志，进行中断 1：收到 ERROR SIGNAL 时根据 T=0 协议进行自动回发。在单一 byte 重复发送次数超过 REP_T 后，置 rx_parity_err 标志，进行中断
输出参数	无

返回值	无
-----	---

### 18.2.28 U7816x\_U7816FRC\_RREPEN\_Getable

函数名	U7816x_U7816FRC_RREPEN_Getable
函数原型	FunState U7816x_U7816FRC_RREPEN_Getable(U7816x_Type* U7816x)
功能描述	获取接收数据奇偶校验错的处理方式选择
输入参数 1	U78160/U78161
输出参数	无
返回值	接收数据奇偶校验错的处理方式选择： 0：收到 ERROR SIGNAL 时不进行自动回发，置 tx_parity_err 标志，进行中断 1：收到 ERROR SIGNAL 时根据 T=0 协议进行自动回发。在单一 byte 重复发送次数超过 REP_T 后，置 rx_parity_err 标志，进行中断

### 18.2.29 U7816x\_U7816FRC\_DICONV\_Setable

函数名	U7816x_U7816FRC_DICONV_Setable
函数原型	void U7816x_U7816FRC_DICONV_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	设置传输次序反向编码使能
输入参数 1	U78160/U78161
输入参数 2	传输次序反向编码使能： DISABLE：禁止传输次序反向编码 ENABLE：使能传输次序反向编码
输出参数	无
返回值	无

### 18.2.30 U7816x\_U7816FRC\_DICONV\_Getable

函数名	U7816x_U7816FRC_DICONV_Getable
函数原型	FunState U7816x_U7816FRC_DICONV_Getable(U7816x_Type* U7816x)
功能描述	获取传输次序反向编码状态
输入参数 1	U78160/U78161
输出参数	无
返回值	传输次序反向编码使能： DISABLE：禁止传输次序反向编码 ENABLE：使能传输次序反向编码

**18.2.31 U7816x\_U7816EGT\_Write**

函数名	U7816x_U7816EGT_Write
函数原型	void U7816x_U7816EGT_Write(U7816x_Type* U7816x, uint32_t SetValue)
功能描述	设置发送时插入的 EGT 时间
输入参数 1	U78160/U78161
输入参数 2	插入的 EGT 时间
输出参数	无
返回值	无

**18.2.32 U7816x\_U7816EGT\_Read**

函数名	U7816x_U7816EGT_Read
函数原型	uint32_t U7816x_U7816EGT_Read(U7816x_Type* U7816x)
功能描述	获取发送时插入的 EGT 时间
输入参数 1	U78160/U78161
输出参数	无
返回值	EGT 寄存器值

**18.2.33 U7816x\_U7816CLKDIV\_Write**

函数名	U7816x_U7816CLKDIV_Write
函数原型	void U7816x_U7816CLKDIV_Write(U7816x_Type* U7816x, uint32_t SetValue)
功能描述	设置 U7816 时钟输出分频控制
输入参数 1	U78160/U78161
输入参数 2	分频控制值:0-1F
输出参数	无
返回值	无

**18.2.34 U7816x\_U7816CLKDIV\_Read**

函数名	U7816x_U7816CLKDIV_Read
函数原型	uint32_t U7816x_U7816CLKDIV_Read(U7816x_Type* U7816x)
功能描述	获取 U7816 时钟输出分频控制值
输入参数 1	U78160/U78161
输出参数	无
返回值	分频控制值:0-1F



**18.2.35 U7816x\_U7816PDIV\_Write**

函数名	U7816x_U7816PDIV_Write
函数原型	void U7816x_U7816PDIV_Write(U7816x_Type* U7816x, uint32_t SetValue)
功能描述	设置 U7816 预分频控制，控制 7816 通信分频比
输入参数 1	U78160/U78161
输入参数 2	预分频值：0-FFF
输出参数	无
返回值	无

**18.2.36 U7816x\_U7816PDIV\_Read**

函数名	U7816x_U7816PDIV_Read
函数原型	uint32_t U7816x_U7816PDIV_Read(U7816x_Type* U7816x)
功能描述	获取 U7816 预分频控制值
输入参数 1	U78160/U78161
输出参数	无
返回值	预分频值：0-FFF

**18.2.37 U7816x\_U7816TXBUF\_Write**

函数名	U7816x_U7816TXBUF_Write
函数原型	void U7816x_U7816TXBUF_Write(U7816x_Type* U7816x, uint32_t SetValue)
功能描述	U7816 发送数据
输入参数 1	U78160/U78161
输入参数 2	发送的数据：0-FF
输出参数	无
返回值	无

**18.2.38 U7816x\_U7816RXBUF\_Read**

函数名	U7816x_U7816RXBUF_Read
函数原型	uint32_t U7816x_U7816RXBUF_Read(U7816x_Type* U7816x)
功能描述	读 U7816 数据接收缓存寄存器
输入参数 1	U78160/U78161
输出参数	无
返回值	接收值：0-FF



**18.2.39 U7816x\_U7816IE\_RXIE\_Setable**

函数名	U7816x_U7816IE_RXIE_Setable
函数原型	void U7816x_U7816IE_RXIE_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	数据接收中断使能位
输入参数 1	U78160/U78161
输入参数 2	数据接收中断使能： DISABLE：禁止数据接收中断 ENABLE：使能数据接收中断
输出参数	无
返回值	无

**18.2.40 U7816x\_U7816IE\_RXIE\_Getable**

函数名	U7816x_U7816IE_RXIE_Getable
函数原型	FunState U7816x_U7816IE_RXIE_Getable(U7816x_Type* U7816x)
功能描述	获取数据接收中断使能位状态
输入参数 1	U78160/U78161
输出参数	无
返回值	数据接收中断使能： DISABLE：禁止数据接收中断 ENABLE：使能数据接收中断

**18.2.41 U7816x\_U7816IE\_TXIE\_Setable**

函数名	U7816x_U7816IE_TXIE_Setable
函数原型	void U7816x_U7816IE_TXIE_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	设置数据发送中断使能
输入参数 1	U78160/U78161
输入参数 2	数据发送中断使能： DISABLE：禁止数据发送中断 ENABLE：使能数据发送中断
输出参数	无
返回值	无

**18.2.42 U7816x\_U7816IE\_TXIE\_Getable**

函数名	U7816x_U7816IE_TXIE_Getable
-----	-----------------------------



函数原型	FunState U7816x_U7816IE_TXIE_Getable(U7816x_Type* U7816x)
功能描述	获取数据发送中断使能状态
输入参数 1	U78160/U78161
输出参数	无
返回值	数据发送中断使能： DISABLE：禁止数据发送中断 ENABLE：使能数据发送中断

#### 18.2.43 U7816x\_U7816IE\_LSIE\_Setable

函数名	U7816x_U7816IE_LSIE_Setable
函数原型	void U7816x_U7816IE_LSIE_Setable(U7816x_Type* U7816x, FunState NewState)
功能描述	设置线路状态中断使能
输入参数 1	U78160/U78161
输入参数 2	线路状态中断使能： DISABLE：禁止线路状态中断 ENABLE：使能线路状态中断
输出参数	无
返回值	无

#### 18.2.44 U7816x\_U7816IE\_LSIE\_Getable

函数名	U7816x_U7816IE_LSIE_Getable
函数原型	FunState U7816x_U7816IE_LSIE_Getable(U7816x_Type* U7816x)
功能描述	获取线路状态中断使能状态
输入参数 1	U78160/U78161
输出参数	无
返回值	线路状态中断使能： DISABLE：禁止线路状态中断 ENABLE：使能线路状态中断

#### 18.2.45 U7816x\_U7816IF\_ERRIF\_Chk

函数名	U7816x_U7816IF_ERRIF_Chk
函数原型	FlagStatus U7816x_U7816IF_ERRIF_Chk(U7816x_Type* U7816x)
功能描述	获取错误中断标志
输入参数 1	U78160/U78161
输出参数	无

返回值	错误中断标志： RESET：未产生错误中断 SET：产生错误中断
-----	----------------------------------------

#### 18.2.46 U7816x\_U7816IF\_TXIF\_Chk

函数名	U7816x_U7816IF_TXIF_Chk
函数原型	FlagStatus U7816x_U7816IF_TXIF_Chk(U7816x_Type* U7816x)
功能描述	获取发送缓冲区空标志
输入参数 1	U78160/U78161
输出参数	无
返回值	发送缓冲区空标志： RESET：发送缓冲区未空 SET：发送缓冲区空

#### 18.2.47 U7816x\_U7816IF\_RXIF\_Chk

函数名	U7816x_U7816IF_RXIF_Chk
函数原型	FlagStatus U7816x_U7816IF_RXIF_Chk(U7816x_Type* U7816x)
功能描述	获取接收完成标志
输入参数 1	U78160/U78161
输出参数	无
返回值	接收完成标志： RESET：未完成接收 SET：完成接收

#### 18.2.48 U7816x\_U7816ERR\_TPARERR\_Clr

函数名	U7816x_U7816ERR_TPARERR_Clr
函数原型	void U7816x_U7816ERR_TPARERR_Clr(U7816x_Type* U7816x)
功能描述	清除发送数据奇偶校验错误标志
输入参数 1	U78160/U78161
输出参数	无
返回值	无

#### 18.2.49 U7816x\_U7816ERR\_TPARERR\_Chk

函数名	U7816x_U7816ERR_TPARERR_Chk
-----	-----------------------------

函数原型	FlagStatus U7816x_U7816ERR_TPARERR_Chk(U7816x_Type* U7816x)
功能描述	获取发送数据奇偶校验错误标志
输入参数 1	U78160/U78161
输出参数	无
返回值	发送数据奇偶校验错误标志： RESET：未产生发送数据奇偶校验错误 SET：产生发送数据奇偶校验错误

### 18.2.50 U7816x\_U7816ERR\_RPARERR\_Clr

函数名	U7816x_U7816ERR_RPARERR_Clr
函数原型	void U7816x_U7816ERR_RPARERR_Clr(U7816x_Type* U7816x)
功能描述	清除接收数据奇偶校验错误标志
输入参数 1	U78160/U78161
输出参数	无
返回值	无

### 18.2.51 U7816x\_U7816ERR\_RPARERR\_Chk

函数名	U7816x_U7816ERR_RPARERR_Chk
函数原型	FlagStatus U7816x_U7816ERR_RPARERR_Chk(U7816x_Type* U7816x)
功能描述	获取接收数据奇偶校验错误标志
输入参数 1	U78160/U78161
输出参数	无
返回值	接收数据奇偶校验错误标志： RESET：未产生接收数据奇偶校验错误 SET：产生接收数据奇偶校验错误

### 18.2.52 U7816x\_U7816ERR\_FRERR\_Clr

函数名	U7816x_U7816ERR_FRERR_Clr
函数原型	void U7816x_U7816ERR_FRERR_Clr(U7816x_Type* U7816x)
功能描述	清除收帧格式错误标志位
输入参数 1	U78160/U78161
输出参数	无
返回值	无

**18.2.53 U7816x\_U7816ERR\_FRERR\_Chk**

函数名	U7816x_U7816ERR_FRERR_Chk
函数原型	FlagStatus U7816x_U7816ERR_FRERR_Chk(U7816x_Type* U7816x)
功能描述	获取接收帧格式错误标志位
输入参数 1	U78160/U78161
输出参数	无
返回值	接收帧格式错误标志： RESET：未产生接收帧格式错误 SET：产生发接收帧格式错误

**18.2.54 U7816x\_U7816ERR\_OVERR\_Clr**

函数名	U7816x_U7816ERR_OVERR_Clr
函数原型	void U7816x_U7816ERR_OVERR_Clr(U7816x_Type* U7816x)
功能描述	清除接收溢出错误标志位
输入参数 1	U78160/U78161
输出参数	无
返回值	无

**18.2.55 U7816x\_U7816ERR\_OVERR\_Chk**

函数名	U7816x_U7816ERR_OVERR_Chk
函数原型	FlagStatus U7816x_U7816ERR_OVERR_Chk(U7816x_Type* U7816x)
功能描述	获取收溢出错误标志位
输入参数 1	U78160/U78161
输入参数 2	无
输出参数	无
返回值	溢出错误标志： RESET：未产生溢出错误 SET：产生发溢出错误

**18.2.56 U7816x\_U7816STA\_WAIT\_RPT\_Chk**

函数名	U7816x_U7816STA_WAIT_RPT_Chk
函数原型	FlagStatus U7816x_U7816STA_WAIT_RPT_Chk(U7816x_Type* U7816x)
功能描述	获取错误等待标志
输入参数 1	U78160/U78161

输出参数	无
返回值	错误等待标志： RESET：未产生错误等待 SET：产生错误等待

### 18.2.57 U7816x\_U7816STA\_TXBUSY\_Chk

函数名	U7816x_U7816STA_TXBUSY_Chk
函数原型	FlagStatus U7816x_U7816STA_TXBUSY_Chk(U7816x_Type* U7816x)
功能描述	获取发送数据忙标志
输入参数 1	U78160/U78161
输入参数 2	无
输出参数	无
返回值	发送数据忙标志： RESET：未产生发送数据忙标志 SET：产生发送数据忙标志

### 18.2.58 U7816x\_U7816STA\_RXBUSY\_Chk

函数名	U7816x_U7816STA_RXBUSY_Chk
函数原型	FlagStatus U7816x_U7816STA_RXBUSY_Chk(U7816x_Type* U7816x)
功能描述	获取接收数据忙标志
输入参数 1	U78160/U78161
输入参数 2	无
输出参数	无
返回值	错误等待标志： RESET：未产生接收数据忙标志 SET：产生接收数据忙标志

## 19 LCD

### 19.1 LCD 寄存器结构

寄存器	描述
DISPCTRL	显示控制寄存器
LCDTEST	显示测试控制寄存器
DF	显示频率控制寄存器
TON	显示点亮时间寄存器
TOFF	显示熄灭时间寄存器
DISPIE	显示中断使能寄存器
DISPIF	显示中断标志寄存器
LCDSET	LCD 显示设置寄存器
LCDDRV	LCD 驱动模式控制寄存器
DISPDATA0	显示数据寄存器 0
DISPDATA1	显示数据寄存器 1
DISPDATA2	显示数据寄存器 2
DISPDATA3	显示数据寄存器 3
DISPDATA4	显示数据寄存器 4
DISPDATA5	显示数据寄存器 5
DISPDATA6	显示数据寄存器 6
DISPDATA7	显示数据寄存器 7
DISPDATA8	显示数据寄存器 8
DISPDATA9	显示数据寄存器 9
LCDBIAS	LCD 显示灰度设置寄存器



COM_EN	LCD COM 使能控制寄存器
SEG_EN0	LCD SEG 使能控制寄存器 0
SEG_EN1	LCD SEG 使能控制寄存器 1
LCDBSTCON	LCDBoost 控制寄存器

## 19.2 LCD 库函数

序号	函数名	描述
1	LCD_Init	LCD 初始化
2	LCD_Deinit	LCD 初始化为缺省值
3	LCD_DISPCTRL_LCDEN_Setable	设置 LCD 显示功能使能控制
4	LCD_DISPCTRL_LCDEN_Getable	获取 LCD 显示功能使能控制
5	LCD_DISPCTRL_FLICK_Setable	设置 LCD 显示闪烁使能控制
6	LCD_DISPCTRL_FLICK_Getable	获取 LCD 显示闪烁功能使能控制
7	LCD_DISPCTRL_TEST_Setable	设置 LCD 测试模式下显示控制
8	LCD_DISPCTRL_TEST_Getable	获取 LCD 测试模式下显示控制
9	LCD_LCDTEST_TESTEN_Setable	设置 LCD 测试模式使能控制
10	LCD_LCDTEST_TESTEN_Getable	获取 LCD 测试模式使能控制
11	LCD_DISPCTRL_DISPMD_Setable	设置 LCD 模式选择
12	LCD_DISPCTRL_DISPMD_Getable	获取 LCD 模式选择
13	LCD_LCDTEST_LCCTRL_Set	设置 LCD 测试控制
14	LCD_LCDTEST_LCCTRL_Get	获取 LCD 测试控制
15	LCD_DF_Write	设置 LCD 显示频率
16	LCD_DF_Read	读取 LCD 显示频率
17	LCD_TON_Write	设置 LCD 显示点亮时间
18	LCD_TON_Read	读取 LCD 显示点亮时间
19	LCD_TOFF_Write	设置 LCD 显示熄灭时间
20	LCD_TOFF_Read	读取 LCD 显示熄灭时间
21	LCD_DISPIE_DONIE_Setable	设置 LCD 点亮中断使能
22	LCD_DISPIE_DONIE_Getable	读取 LCD 点亮中断使能
23	LCD_DISPIE_DOFFIE_Setable	设置 LCD 熄灭中断使能
24	LCD_DISPIE_DOFFIE_Getable	读取 LCD 熄灭中断使能
25	LCD_DISPIF_DONIF_Clr	清 LCD 点亮中断标志
26	LCD_DISPIF_DONIF_Chk	获取 LCD 点亮中断标志
27	LCD_DISPIF_DOFFIF_Clr	清 LCD 熄灭中断标志





28	LCD_DISPIF_DOFFIF_Chk	获取 LCD 熄灭中断标志
29	LCD_LCDSET_BIASMD_Set	设置 LCD 偏置
30	LCD_LCDSET_BIASMD_Get	获取 LCD 偏置
31	LCD_LCDSET_WFT_Set	设置 LCD 驱动类型
32	LCD_LCDSET_WFT_Get	获取 LCD 驱动类型
33	LCD_LCDSET_LMUX_Set	设置 LCD duty 值
34	LCD_LCDSET_LMUX_Get	获取 LCD duty 值
35	LCD_LCDDRV_SCFSEL_Set	设置 LCD SC 频率值
36	LCD_LCDDRV_SCFSEL_Get	获取 LCD SC 频率值
37	LCD_LCDDRV_SC_CTRL_Set	设置 LCD 驱动方式控制
38	LCD_LCDDRV_SC_CTRL_Get	获取 LCD 驱动方式控制
39	LCD_LCDDRV_IC_CTRL_Set	设置 LCD 驱动电流
40	LCD_LCDDRV_IC_CTRL_Get	获取 LCD 驱动电流
41	LCD_LCDDRV_ENMODE_Set	设置 LCD 驱动模式
42	LCD_LCDDRV_ENMODE_Get	获取 LCD 驱动模式
43	LCD_LCDBIAS_Write	设置 LCD 偏置电平选择
44	LCD_LCDBIAS_Read	获取 LCD 偏置电平选择
45	LCD_LCDBSTCON_OSC_CFG_Set	设置 LCD BOOST 环振频率设置信号
46	LCD_LCDBSTCON_OSC_CFG_Get	获取 LCD BOOST 环振频率设置信号
47	LCD_LCDBSTCON_BUFENB_Setable	设置 LCD 基准源 Buffer 使能控制
48	LCD_LCDBSTCON_BUFENB_Getable	获取 LCD 基准源 Buffer 使能控制
49	LCD_LCDBSTCON_BUFBYP_Setable	设置 LCD 基准源 Buffer 旁路控制
50	LCD_LCDBSTCON_BUFBYP_Getable	获取 LCD 基准源 Buffer 旁路控制
51	LCD_LCDBSTCON_VLCDPD_Setable	设置 LCD VLCD 下拉
52	LCD_LCDBSTCON_VLCDPD_Getable	获取 LCD VLCD 下拉状态
53	LCD_LCDBSTCON_VLCDCFG_Set	设置 LCD VLCD 输出电压
54	LCD_LCDBSTCON_VLCDCFG_Get	获取 LCD VLCD 输出电压
55	LCD_LCDBSTCON_BOOST_TEN_Setable	设置 LCD Booster 测试信号
56	LCD_LCDBSTCON_BOOST_TEN_Getable	获取 LCD Booster 测试信号状态
57	LCD_LCDBSTCON_BOOST_EN_Setable	设置 LCD Booster 使能控制



58	LCD_LCDBSTCON_BOOST_EN_Getable	获取 LCD Booster 使能状态
59	LCD_COM_EN_Write	设置 LCD COM0~3 输出使能控制
60	LCD_COM_EN_Read	获取 LCD COM0~3 输出使能状态
61	LCD_SEG_EN0_Write	设置 LCD SEG0-31 输出使能控制
62	LCD_SEG_EN0_Read	获取 LCD SEG0-31 输出使能状态
63	LCD_SEG_EN1_Write	设置 LCD SEG32~43(COM4-7)输出使能控制
64	LCD_SEG_EN1_Read	获取 LCD SEG32~43(COM4-7)输出使能状态
65	LCD_DISPDATAx_Refresh	设置 DISPDATAx 显示数据寄存器刷新
66	LCD_Boost_Init	ICDBoost 功能初始化
67	LCD_DISPCTRL_ANTIPOLAR_Setable	设置 SEG/COM 在 LCD 禁止情况下接地控制, 防极化
68	LCD_DISPCTRL_ANTIPOLAR_Getable	获取 SEG/COM 在 LCD 禁止情况下接地状态

### 19.2.1 LCD\_Init

函数名	LCD_Init
函数原型	void LCD_Init(LCD_InitTypeDef* para)
功能描述	LCD 初始化
输入参数 1	<div>LCD 参数:</div> <div><div>COM 数量选择: 00: 4COM 01: 6COM 10/11: 8COM</div><div>驱动模式选择: 0: 片外电容驱动 1: 片内电阻驱动</div><div>驱动波形选择: 0: A 类波形 1: B 类波形</div></div>

显示预分频 DF(0-FF)

DF≠0 时显示扫描频率计算:

COM 数量	帧格式 Hz	
	A 类波形	B 类波形
4	显示电路工作频率/ (4*DF*2)	显示电路工作频率/ (4*DF*4)
6	显示电路工作频率/ (6*DF*2)	显示电路工作频率/ (6*DF*4)
8	显示电路工作频率/ (8*DF*2)	显示电路工作频率/ (8*DF*4)

DF=0 显示时扫描频率计算:

COM 数量	帧格式 Hz	
	A 类波形	B 类波形
4	显示电路工作频率/4	显示电路工作频率/ (4*2)
6	显示电路工作频率/6	显示电路工作频率/ (6* 2)
8	显示电路工作频率/8	显示电路工作频率/ (8* 2)

偏置类型控制:

LCD\_LCDSET\_BIASMD\_4BIAS: 1/4bias

LCD\_LCDSET\_BIASMD\_3BIAS: 1/3bias

SC 频率选择:

LCD\_LCDDRV\_SCFSEL\_X1: 频率为 disp\_clk/1

LCD\_LCDDRV\_SCFSEL\_X8: 频率为 disp\_clk/8

LCD\_LCDDRV\_SCFSEL\_X16: 频率为 disp\_clk/16

LCD\_LCDDRV\_SCFSEL\_X32: 频率为 disp\_clk/32

LCD\_LCDDRV\_SCFSEL\_X64: 频率为 disp\_clk/64

LCD\_LCDDRV\_SCFSEL\_X128: 频率为 disp\_clk/128

LCD\_LCDDRV\_SCFSEL\_X256: 频率为 disp\_clk/256

LCD\_LCDDRV\_SCFSEL\_X512: 频率为 disp\_clk/512

片外电容驱动模式下, 驱动方式控制:

LCD\_LCDDRV\_SC\_CTRL\_ONE: 单次驱动

LCD\_LCDDRV\_SC\_CTRL\_TWO: 连续驱动两次

LCD\_LCDDRV\_SC\_CTRL\_FOUR: 连续驱动四次(当 SC&gt;4Hz 时, 此选项也为连续驱动)

LCD\_LCDDRV\_SC\_CTRL\_CONTINUE: 连续驱动

片内电阻模式下, 驱动电流大小:

LCD\_LCDDRV\_IC\_CTRL\_L3: 驱动电流最大

LCD\_LCDDRV\_IC\_CTRL\_L2: 驱动电流次大

LCD\_LCDDRV\_IC\_CTRL\_L1: 驱动电流次小

LCD\_LCDDRV\_IC\_CTRL\_L0: 驱动电流最小

显示灰度设置:

SVDLVL	不同 VDD 下内部偏置电压 V			
	5	4.5	3.6	3.1
0000	2.74	2.47	1.97	1.64
0001	2.83	2.54	2.03	1.69
0010	2.92	2.62	2.10	1.75
0011	3.01	2.71	2.17	1.81
0100	3.12	2.80	2.24	1.87

	<table><tr><td>0101</td><td>3.23</td><td>2.90</td><td>2.32</td><td>1.94</td></tr><tr><td>0110</td><td>3.35</td><td>3.01</td><td>2.41</td><td>2.01</td></tr><tr><td>0111</td><td>3.47</td><td>3.13</td><td>2.50</td><td>2.08</td></tr><tr><td>1000</td><td>3.61</td><td>3.25</td><td>2.60</td><td>2.17</td></tr><tr><td>1001</td><td>3.76</td><td>3.39</td><td>2.71</td><td>2.26</td></tr><tr><td>1010</td><td>3.93</td><td>3.53</td><td>2.83</td><td>2.35</td></tr><tr><td>1011</td><td>4.10</td><td>3.69</td><td>2.95</td><td>2.46</td></tr><tr><td>1100</td><td>4.30</td><td>3.87</td><td>3.09</td><td>2.58</td></tr><tr><td>1101</td><td>5.51</td><td>4.06</td><td>3.25</td><td>2.71</td></tr><tr><td>1110</td><td>4.75</td><td>4.27</td><td>3.42</td><td>2.85</td></tr><tr><td>1111</td><td>5.00</td><td>4.50</td><td>3.60</td><td>3.00</td></tr></table>	0101	3.23	2.90	2.32	1.94	0110	3.35	3.01	2.41	2.01	0111	3.47	3.13	2.50	2.08	1000	3.61	3.25	2.60	2.17	1001	3.76	3.39	2.71	2.26	1010	3.93	3.53	2.83	2.35	1011	4.10	3.69	2.95	2.46	1100	4.30	3.87	3.09	2.58	1101	5.51	4.06	3.25	2.71	1110	4.75	4.27	3.42	2.85	1111	5.00	4.50	3.60	3.00
0101	3.23	2.90	2.32	1.94																																																				
0110	3.35	3.01	2.41	2.01																																																				
0111	3.47	3.13	2.50	2.08																																																				
1000	3.61	3.25	2.60	2.17																																																				
1001	3.76	3.39	2.71	2.26																																																				
1010	3.93	3.53	2.83	2.35																																																				
1011	4.10	3.69	2.95	2.46																																																				
1100	4.30	3.87	3.09	2.58																																																				
1101	5.51	4.06	3.25	2.71																																																				
1110	4.75	4.27	3.42	2.85																																																				
1111	5.00	4.50	3.60	3.00																																																				
	<p>防极化使能控制： 0: COM 和 SEG 在 LCD 关闭情况下浮空 1: COM 和 SEG 在 LCD 关闭情况下接地</p>																																																							
	<p>测试模式使能控制，仅在 DISPMD=1 的情况下有效： DISABLE：禁止 LCD 测试模式 ENABLE：使能 LCD 测试模式</p>																																																							
	<p>测试模式选择控制： 0: 正常模式 1: 显示测试模式</p>																																																							
	<p>LCD 测试控制位，保持为 0</p>																																																							
	<p>测试模式使能，保持禁止： DISABLE：禁止测试模式 ENABLE：使能测试模式</p>																																																							
	<p>显示闪烁使能： DISABLE：禁止显示闪烁 ENABLE：使能显示闪烁</p>																																																							
	<p>闪烁显示时的点亮时间 = TON×0.25 秒</p>																																																							
	<p>闪烁显示时的熄灭时间 = TOFF×0.25 秒</p>																																																							
	<p>显示点亮中断使能控制： DISABLE：禁止点亮中断 ENABLE：使能点亮中断</p>																																																							
	<p>显示熄灭中断使能控制： DISABLE：禁止熄灭中断 ENABLE：使能熄灭中断</p>																																																							
	<p>LCD 显示使能控制： DISABLE：禁止 LCD 显示 ENABLE：使能 LCD 显示</p>																																																							
输出参数	无																																																							
返回值	无																																																							

### 19.2.2 LCD\_Deinit

函数名	LCD_Deinit
函数原型	void LCD_Deinit(void)
功能描述	LCD 初始化为缺省值
输入参数 1	无
输出参数	无
返回值	无

### 19.2.3 LCD\_DISPCTRL\_LCDEN\_Setable

函数名	LCD_DISPCTRL_LCDEN_Setable
函数原型	void LCD_DISPCTRL_LCDEN_Setable(FunState NewState)
功能描述	设置 LCD 显示功能使能控制
输入参数 1	LCD 显示功能使能控制： DISABLE: 禁止 LCD 显示 ENABLE: 使能 LCD 显示
输出参数	无
返回值	无

### 19.2.4 LCD\_DISPCTRL\_LCDEN\_Getable

函数名	LCD_DISPCTRL_LCDEN_Getable
函数原型	FunState LCD_DISPCTRL_LCDEN_Getable(void)
功能描述	获取 LCD 显示功能使能控制
输入参数 1	无
输出参数	无
返回值	LCD 显示功能使能控制： DISABLE: 禁止 LCD 显示 ENABLE: 使能 LCD 显示

### 19.2.5 LCD\_DISPCTRL\_FLICK\_Setable

函数名	LCD_DISPCTRL_FLICK_Setable
函数原型	void LCD_DISPCTRL_FLICK_Setable(FunState NewState)
功能描述	设置 LCD 显示闪烁使能控制
输入参数 1	LCD 显示闪烁使能控制： DISABLE: 禁止 LCD 显示闪烁

	ENABLE: 使能 LCD 显示闪烁
输出参数	无
返回值	无

### 19.2.6 LCD\_DISPCTRL\_FLICK\_Getable

函数名	LCD_DISPCTRL_FLICK_Getable
函数原型	FunState LCD_DISPCTRL_FLICK_Getable(void)
功能描述	获取 LCD 显示闪烁功能使能控制
输入参数 1	无
输出参数	无
返回值	LCD 显示闪烁使能控制: DISABLE: 禁止 LCD 显示闪烁 ENABLE: 使能 LCD 显示闪烁

### 19.2.7 LCD\_DISPCTRL\_TEST\_Setable

函数名	LCD_DISPCTRL_TEST_Setable
函数原型	void LCD_DISPCTRL_TEST_Setable(FunState NewState)
功能描述	设置 LCD 测试模式下显示控制(仅在 DISPMD=1 的情况下有效)
输入参数 1	LCD 测试模式下显示控制: DISABLE: LCD 测试模式下全灭 ENABLE: LCD 测试模式下全亮
输出参数	无
返回值	无

### 19.2.8 LCD\_DISPCTRL\_TEST\_Getable

函数名	LCD_DISPCTRL_TEST_Getable
函数原型	FunState LCD_DISPCTRL_TEST_Getable(void)
功能描述	获取 LCD 测试模式下显示控制(仅在 DISPMD=1 的情况下有效)
输入参数 1	无
输出参数	无
返回值	LCD 测试模式下显示控制: DISABLE: LCD 测试模式下全灭 ENABLE: LCD 测试模式下全亮

### 19.2.9 LCD\_LCDTEST\_TESTEN\_Setable

函数名	LCD_LCDTEST_TESTEN_Setable
函数原型	void LCD_LCDTEST_TESTEN_Setable(FunState NewState)
功能描述	设置 LCD 测试模式使能控制 (仅在 DISPMD=1 的情况下有效)
输入参数 1	LCD 测试模式使能控制: DISABLE: 禁止 LCD 测试模式 ENABLE: 使能 LCD 测试模式
输出参数	无
返回值	无

### 19.2.10 LCD\_LCDTEST\_TESTEN\_Getable

函数名	LCD_LCDTEST_TESTEN_Getable
函数原型	FunState LCD_LCDTEST_TESTEN_Getable(void)
功能描述	获取 LCD 测试模式使能控制(仅在 DISPMD=1 的情况下有效)
输入参数 1	无
输出参数	无
返回值	LCD 测试模式使能控制: DISABLE: 禁止 LCD 测试模式 ENABLE: 使能 LCD 测试模式

### 19.2.11 LCD\_DISPCTRL\_DISPMD\_Setable

函数名	LCD_DISPCTRL_DISPMD_Setable
函数原型	void LCD_DISPCTRL_DISPMD_Setable(FunState NewState)
功能描述	设置 LCD 模式选择
输入参数 1	LCD 模式选择: DISABLE: 正常模式 ENABLE: 测试模式
输出参数	无
返回值	无

### 19.2.12 LCD\_DISPCTRL\_DISPMD\_Getable

函数名	LCD_DISPCTRL_DISPMD_Getable
函数原型	FunState LCD_DISPCTRL_DISPMD_Getable(void)
功能描述	获取 LCD 模式选择



输入参数 1	无
输出参数	无
返回值	LCD 模式选择: DISABLE: 正常模式 ENABLE: 测试模式

### 19.2.13 LCD\_LCDTEST\_LCCTRL\_Set

函数名	LCD_LCDTEST_LCCTRL_Set
函数原型	void LCD_LCDTEST_LCCTRL_Set(uint32_t SetValue)
功能描述	设置 LCD 测试控制(仅在 DISPMD=1 的情况下有效)
输入参数 1	LCD 测试控制: LCD_LCDTEST_LCCTRL_0: LCD 电平由正常寄存器设置 LCD_LCDTEST_LCCTRL_1: LCD 电平由测试寄存器设置
输出参数	无
返回值	无

### 19.2.14 LCD\_LCDTEST\_LCCTRL\_Get

函数名	LCD_LCDTEST_LCCTRL_Get
函数原型	uint32_t LCD_LCDTEST_LCCTRL_Get(void)
功能描述	获取 LCD 测试控制
输入参数 1	无
输出参数	无
返回值	LCD 测试控制: LCD_LCDTEST_LCCTRL_0: LCD 电平由正常寄存器设置 LCD_LCDTEST_LCCTRL_1: LCD 电平由测试寄存器设置

### 19.2.15 LCD\_DF\_Write

函数名	LCD_DF_Write
函数原型	void LCD_DF_Write(uint32_t SetValue)
功能描述	配置 LCD 显示频率
输入参数 1	显示预分频值: 0-FF
输出参数	无
返回值	无



**19.2.16 LCD\_DF\_Read**

函数名	LCD_DF_Read
函数原型	uint32_t LCD_DF_Read(void)
功能描述	读取 LCD 显示频率
输入参数 1	无
输出参数	无
返回值	显示预分频值：0-FF

**19.2.17 LCD\_TON\_Write**

函数名	LCD_TON_Write
函数原型	void LCD_TON_Write(uint32_t SetValue)
功能描述	配置 LCD 显示点亮时间
输入参数 1	LCD 显示点亮时间：00-FF ton*0.25s
输出参数	无
返回值	无

**19.2.18 LCD\_TON\_Read**

函数名	LCD_TON_Read
函数原型	uint32_t LCD_TON_Read(void)
功能描述	读取 LCD 显示点亮时间
输入参数 1	无
输出参数	无
返回值	LCD 显示点亮时间：00-FF ton*0.25s

**19.2.19 LCD\_TOFF\_Write**

函数名	LCD_TOFF_Write
函数原型	void LCD_TOFF_Write(uint32_t SetValue)
功能描述	配置 LCD 显示熄灭时间
输入参数 1	LCD 显示熄灭时间：00-FF ton*0.25s
输出参数	无
返回值	无

**19.2.20 LCD\_TOFF\_Read**

函数名	LCD_TOFF_Read
函数原型	uint32_t LCD_TOFF_Read(void)
功能描述	读取 LCD 显示熄灭时间
输入参数 1	无
输出参数	无
返回值	LCD 显示熄灭时间: 00-FF ton*0.25s

**19.2.21 LCD\_DISPIE\_DONIE\_Setable**

函数名	LCD_DISPIE_DONIE_Setable
函数原型	void LCD_DISPIE_DONIE_Setable(FunState NewState)
功能描述	配置 LCD 点亮中断使能
输入参数 1	LCD 点亮中断使能: DISABLE: 禁止点亮中断 ENABLE: 使能点亮中断
输出参数	无
返回值	无

**19.2.22 LCD\_DISPIE\_DONIE\_Getable**

函数名	LCD_DISPIE_DONIE_Getable
函数原型	FunState LCD_DISPIE_DONIE_Getable(void)
功能描述	读取 LCD 点亮中断使能
输入参数 1	无
输出参数	无
返回值	LCD 点亮中断使能: DISABLE: 禁止点亮中断 ENABLE: 使能点亮中断

**19.2.23 LCD\_DISPIE\_DOFFIE\_Setable**

函数名	LCD_DISPIE_DOFFIE_Setable
函数原型	void LCD_DISPIE_DOFFIE_Setable(FunState NewState)
功能描述	配置 LCD 熄灭中断使能
输入参数 1	LCD 熄灭中断使能: DISABLE: 禁止熄灭中断

	ENABLE: 使能熄灭中断
输出参数	无
返回值	无

#### 19.2.24 LCD\_DISPIE\_DOFFIE\_Getable

函数名	LCD_DISPIE_DOFFIE_Getable
函数原型	FunState LCD_DISPIE_DOFFIE_Getable(void)
功能描述	读取 LCD 熄灭中断使能
输入参数 1	无
输出参数	无
返回值	LCD 熄灭中断使能: DISABLE: 禁止熄灭中断 ENABLE: 使能熄灭中断

#### 19.2.25 LCD\_DISPIF\_DONIF\_Clr

函数名	LCD_DISPIF_DONIF_Clr
函数原型	void LCD_DISPIF_DONIF_Clr(void)
功能描述	清 LCD 点亮中断标志
输入参数 1	无
输出参数	无
返回值	无

#### 19.2.26 LCD\_DISPIF\_DONIF\_Chk

函数名	LCD_DISPIF_DONIF_Chk
函数原型	FlagStatus LCD_DISPIF_DONIF_Chk(void)
功能描述	获取 LCD 点亮中断标志
输入参数 1	无
输出参数	无
返回值	LCD 点亮中断标志: RESET: 未发生 LCD 点亮中断 SET: 发生 LCD 点亮中断

#### 19.2.27 LCD\_DISPIF\_DOFFIF\_Clr

函数名	LCD_DISPIF_DOFFIF_Clr
-----	-----------------------

函数原型	void LCD_DISPIF_DOFFIF_Clr(void)
功能描述	清 LCD 熄灭中断标志
输入参数 1	无
输出参数	无
返回值	无

### 19.2.28 LCD\_DISPIF\_DOFFIF\_Chk

函数名	LCD_DISPIF_DOFFIF_Chk
函数原型	FlagStatus LCD_DISPIF_DOFFIF_Chk(void)
功能描述	获取 LCD 熄灭中断标志
输入参数 1	无
输出参数	无
返回值	LCD 熄灭中断标志: RESET: 未发生 LCD 熄灭中断 SET: 发生 LCD 熄灭中断

### 19.2.29 LCD\_LCDSET\_BIASMD\_Set

函数名	LCD_LCDSET_BIASMD_Set
函数原型	void LCD_LCDSET_BIASMD_Set(uint32_t SetValue)
功能描述	设置 LCD 偏置
输入参数 1	LCD 偏置类型: LCD_LCDSET_BIASMD_4BIAS: 1/4bias LCD_LCDSET_BIASMD_3BIAS: 1/3bias
输出参数	无
返回值	无

### 19.2.30 LCD\_LCDSET\_BIASMD\_Get

函数名	LCD_LCDSET_BIASMD_Get
函数原型	uint32_t LCD_LCDSET_BIASMD_Get(void)
功能描述	获取 LCD 偏置
输入参数 1	无
输出参数	无
返回值	LCD 偏置类型: LCD_LCDSET_BIASMD_4BIAS: 1/4bias LCD_LCDSET_BIASMD_3BIAS: 1/3bias

### 19.2.31 LCD\_LCDSET\_WFT\_Set

函数名	LCD_LCDSET_WFT_Set
函数原型	void LCD_LCDSET_WFT_Set(uint32_t SetValue)
功能描述	设置 LCD 驱动类型
输入参数 1	LCD 驱动类型: LCD_LCDSET_WFT_ATYPE: A 类波形 LCD_LCDSET_WFT_BTTYPE: B 类波形
输出参数	无
返回值	无

### 19.2.32 LCD\_LCDSET\_WFT\_Get

函数名	LCD_LCDSET_WFT_Get
函数原型	uint32_t LCD_LCDSET_WFT_Get(void)
功能描述	获取 LCD 驱动类型
输入参数 1	无
输出参数	无
返回值	LCD 驱动类型: LCD_LCDSET_WFT_ATYPE: A 类波形 LCD_LCDSET_WFT_BTTYPE: B 类波形

### 19.2.33 LCD\_LCDSET\_LMUX\_Set

函数名	LCD_LCDSET_LMUX_Set
函数原型	void LCD_LCDSET_LMUX_Set(uint32_t SetValue)
功能描述	设置 LCD duty 值
输入参数 1	LCD duty 值: LCD_LCDSET_LMUX_4COM: 4com LCD_LCDSET_LMUX_6COM: 6com LCD_LCDSET_LMUX_8COM: 8com
输出参数	无
返回值	无

### 19.2.34 LCD\_LCDSET\_LMUX\_Get

函数名	LCD_LCDSET_LMUX_Get
-----	---------------------

函数原型	uint32_t LCD_LCDSET_LMUX_Get(void)
功能描述	获取 LCD duty 值
输入参数 1	无
输出参数	无
返回值	LCD duty 值: LCD_LCDSET_LMUX_4COM: 4com LCD_LCDSET_LMUX_6COM: 6com LCD_LCDSET_LMUX_8COM: 8com

### 19.2.35 LCD\_LCDDRV\_SCFSEL\_Set

函数名	LCD_LCDDRV_SCFSEL_Set
函数原型	void LCD_LCDDRV_SCFSEL_Set(uint32_t SetValue)
功能描述	设置 LCD SC 频率值
输入参数 1	LCD SC 频率值: LCD_LCDDRV_SCFSEL_X1: 频率为 disp_clk/1 LCD_LCDDRV_SCFSEL_X8: 频率为 disp_clk/8 LCD_LCDDRV_SCFSEL_X16: 频率为 disp_clk/16 LCD_LCDDRV_SCFSEL_X32: 频率为 disp_clk/32 LCD_LCDDRV_SCFSEL_X64: 频率为 disp_clk/64 LCD_LCDDRV_SCFSEL_X128: 频率为 disp_clk/128 LCD_LCDDRV_SCFSEL_X256: 频率为 disp_clk/256 LCD_LCDDRV_SCFSEL_X512: 频率为 disp_clk/512
输出参数	无
返回值	无

### 19.2.36 LCD\_LCDDRV\_SCFSEL\_Get

函数名	LCD_LCDDRV_SCFSEL_Get
函数原型	uint32_t LCD_LCDDRV_SCFSEL_Get(void)
功能描述	获取 LCD SC 频率值
输入参数 1	无
输出参数	无
返回值	LCD SC 频率值: LCD_LCDDRV_SCFSEL_X1: 频率为 disp_clk/1 LCD_LCDDRV_SCFSEL_X8: 频率为 disp_clk/8 LCD_LCDDRV_SCFSEL_X16: 频率为 disp_clk/16 LCD_LCDDRV_SCFSEL_X32: 频率为 disp_clk/32

	LCD_LCDDRV_SCFSEL_X64: 频率为 disp_clk/64 LCD_LCDDRV_SCFSEL_X128: 频率为 disp_clk/128 LCD_LCDDRV_SCFSEL_X256: 频率为 disp_clk/256 LCD_LCDDRV_SCFSEL_X512: 频率为 disp_clk/512
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 19.2.37 LCD\_LCDDRV\_SC\_CTRL\_Set

函数名	LCD_LCDDRV_SC_CTRL_Set
函数原型	void LCD_LCDDRV_SC_CTRL_Set(uint32_t SetValue)
功能描述	设置 LCD 驱动方式控制
输入参数 1	LCD 驱动方式控制: LCD_LCDDRV_SC_CTRL_ONE: 单次驱动 LCD_LCDDRV_SC_CTRL_TWO: 连续驱动两次 LCD_LCDDRV_SC_CTRL_FOUR: 连续驱动四次(当 SC>4Hz 时, 此选项也为连续驱动) LCD_LCDDRV_SC_CTRL_CONTINUE: 连续驱动
输出参数	无
返回值	无

### 19.2.38 LCD\_LCDDRV\_SC\_CTRL\_Get

函数名	LCD_LCDDRV_SC_CTRL_Get
函数原型	uint32_t LCD_LCDDRV_SC_CTRL_Get(void)
功能描述	获取 LCD 驱动方式控制
输入参数 1	无
输出参数	无
返回值	LCD 驱动方式控制: LCD_LCDDRV_SC_CTRL_ONE: 单次驱动 LCD_LCDDRV_SC_CTRL_TWO: 连续驱动两次 LCD_LCDDRV_SC_CTRL_FOUR: 连续驱动四次(当 SC>4Hz 时, 此选项也为连续驱动) LCD_LCDDRV_SC_CTRL_CONTINUE: 连续驱动

### 19.2.39 LCD\_LCDDRV\_IC\_CTRL\_Set

函数名	LCD_LCDDRV_IC_CTRL_Set
函数原型	void LCD_LCDDRV_IC_CTRL_Set(uint32_t SetValue)
功能描述	设置 LCD 驱动电流
输入参数 1	LCD 驱动电流: LCD_LCDDRV_IC_CTRL_L3: 驱动电流最大

	LCD_LCDDRV_IC_CTRL_L2: 驱动电流次大 LCD_LCDDRV_IC_CTRL_L1: 驱动电流次小 LCD_LCDDRV_IC_CTRL_L0: 驱动电流最小
输出参数	无
返回值	无

#### 19.2.40 LCD\_LCDDRV\_IC\_CTRL\_Get

函数名	LCD_LCDDRV_IC_CTRL_Get
函数原型	uint32_t LCD_LCDDRV_IC_CTRL_Get(void)
功能描述	获取 LCD 驱动电流
输入参数 1	无
输出参数	无
返回值	LCD 驱动电流: LCD_LCDDRV_IC_CTRL_L3: 驱动电流最大 LCD_LCDDRV_IC_CTRL_L2: 驱动电流次大 LCD_LCDDRV_IC_CTRL_L1: 驱动电流次小 LCD_LCDDRV_IC_CTRL_L0: 驱动电流最小

#### 19.2.41 LCD\_LCDDRV\_ENMODE\_Set

函数名	LCD_LCDDRV_ENMODE_Set
函数原型	void LCD_LCDDRV_ENMODE_Set(uint32_t SetValue)
功能描述	设置 LCD 驱动模式
输入参数 1	LCD 驱动模式: LCD_LCDDRV_ENMODE_EXTERNALCAP: 片外电容驱动 LCD_LCDDRV_ENMODE_INNERRESISTER: 片内电阻驱动
输出参数	无
返回值	无

#### 19.2.42 LCD\_LCDDRV\_ENMODE\_Get

函数名	LCD_LCDDRV_ENMODE_Get
函数原型	uint32_t LCD_LCDDRV_ENMODE_Get(void)
功能描述	获取 LCD 驱动模式
输入参数 1	无
输出参数	无
返回值	LCD 驱动模式:



LCD\_LCDDRV\_ENMODE\_EXTERNALCAP: 片外电容驱动  
LCD\_LCDDRV\_ENMODE\_INNERRESISTER: 片内电阻驱动

### 19.2.43 LCD\_LCDBIAS\_Write

函数名	LCD_LCDBIAS_Write				
函数原型	void LCD_LCDBIAS_Write(uint32_t SetValue)				
功能描述	设置 LCD 偏置电平选择				
输入参数 1	LCD 偏置电平选择:				
	SVDLVL	不同 VDD 下内部偏置电压 V			
		5	4.5	3.6	3.1
	0000	2.74	2.47	1.97	1.64
	0001	2.83	2.54	2.03	1.69
	0010	2.92	2.62	2.10	1.75
	0011	3.01	2.71	2.17	1.81
	0100	3.12	2.80	2.24	1.87
	0101	3.23	2.90	2.32	1.94
	0110	3.35	3.01	2.41	2.01
	0111	3.47	3.13	2.50	2.08
	1000	3.61	3.25	2.60	2.17
	1001	3.76	3.39	2.71	2.26
	1010	3.93	3.53	2.83	2.35
	1011	4.10	3.69	2.95	2.46
	1100	4.30	3.87	3.09	2.58
	1101	5.51	4.06	3.25	2.71
	1110	4.75	4.27	3.42	2.85
	1111	5.00	4.50	3.60	3.00
输出参数	无				
返回值	无				

### 19.2.44 LCD\_LCDBIAS\_Read

函数名	LCD_LCDBIAS_Read				
函数原型	uint32_t LCD_LCDBIAS_Read(void)				
功能描述	获取 LCD 偏置电平选择				
输入参数 1	无				
输出参数	无				
返回值	LCD 偏置电平选择:				
	SVDLVL	不同 VDD 下内部偏置电压 V			
		5	4.5	3.6	3.1

	0000	2.74	2.47	1.97	1.64
	0001	2.83	2.54	2.03	1.69
	0010	2.92	2.62	2.10	1.75
	0011	3.01	2.71	2.17	1.81
	0100	3.12	2.80	2.24	1.87
	0101	3.23	2.90	2.32	1.94
	0110	3.35	3.01	2.41	2.01
	0111	3.47	3.13	2.50	2.08
	1000	3.61	3.25	2.60	2.17
	1001	3.76	3.39	2.71	2.26
	1010	3.93	3.53	2.83	2.35
	1011	4.10	3.69	2.95	2.46
	1100	4.30	3.87	3.09	2.58
	1101	5.51	4.06	3.25	2.71
	1110	4.75	4.27	3.42	2.85
	1111	5.00	4.50	3.60	3.00

#### 19.2.45 LCD\_LCDBSTCON\_OSC\_CFG\_Set

函数名	LCD_LCDBSTCON_OSC_CFG_Set
函数原型	void LCD_LCDBSTCON_OSC_CFG_Set(uint32_t SetValue)
功能描述	设置 LCD BOOST 环振频率配置
输入参数 1	LCD BOOST 环振频率： LCD_LCDBSTCON_OSC_CFG_GRADE0: 0 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE1: 1 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE2: 2 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE3: 3 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE4: 4 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE5: 5 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE6: 6 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE7: 7 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE8: 8 级 值越大频率越高，驱动越强，功耗越高
输出参数	无
返回值	无

#### 19.2.46 LCD\_LCDBSTCON\_OSC\_CFG\_Get

函数名	LCD_LCDBSTCON_OSC_CFG_Get
函数原型	uint32_t LCD_LCDBSTCON_OSC_CFG_Get(void)
功能描述	获取 LCD BOOST 环振频率配置信号

输入参数 1	无
输出参数	无
返回值	LCD BOOST 环振频率： LCD_LCDBSTCON_OSC_CFG_GRADE0: 0 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE1: 1 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE2: 2 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE3: 3 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE4: 4 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE5: 5 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE6: 6 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE7: 7 级 值越大频率越高，驱动越强，功耗越高 LCD_LCDBSTCON_OSC_CFG_GRADE8: 8 级 值越大频率越高，驱动越强，功耗越高

#### 19.2.47 LCD\_LCDBSTCON\_BUFENB\_Setable

函数名	LCD_LCDBSTCON_BUFENB_Setable
函数原型	void LCD_LCDBSTCON_BUFENB_Setable(FunState NewState)
功能描述	配置 LCD 基准源 Buffer 使能控制
输入参数 1	LCD 基准源 Buffer 使能控制： DISABLE: 禁止基准源 ENABLE: 使能基准源
输出参数	无
返回值	无

#### 19.2.48 LCD\_LCDBSTCON\_BUFENB\_Getable

函数名	LCD_LCDBSTCON_BUFENB_Getable
函数原型	FunState LCD_LCDBSTCON_BUFENB_Getable(void)
功能描述	获取 LCD 基准源 Buffer 使能控制
输入参数 1	无
输出参数	无
返回值	LCD 基准源 Buffer 使能控制： DISABLE: 禁止基准源 ENABLE: 使能基准源

#### 19.2.49 LCD\_LCDBSTCON\_BUFBY\_P\_Setable

函数名	LCD_LCDBSTCON_BUFBY_P_Setable
-----	-------------------------------

函数原型	void LCD_LCDBSTCON_BUFBY_P_Setable(FunState NewState)
功能描述	配置 LCD 基准源 Buffer 旁路控制
输入参数 1	LCD 基准源 Buffer 旁路控制： DISABLE：禁止基准源旁路 ENABLE：使能基准源旁路
输出参数	无
返回值	无

### 19.2.50 LCD\_LCDBSTCON\_BUFBY\_P\_Getable

函数名	LCD_LCDBSTCON_BUFBY_P_Getable
函数原型	FunState LCD_LCDBSTCON_BUFBY_P_Getable(void)
功能描述	获取 LCD 基准源 Buffer 旁路控制
输入参数 1	无
输出参数	无
返回值	LCD 基准源 Buffer 旁路控制： DISABLE：禁止基准源旁路 ENABLE：使能基准源旁路

### 19.2.51 LCD\_LCDBSTCON\_VLCDPD\_Setable

函数名	LCD_LCDBSTCON_VLCDPD_Setable
函数原型	void LCD_LCDBSTCON_VLCDPD_Setable(FunState NewState)
功能描述	配置 LCD VLCD 下拉
输入参数 1	LCD VLCD 下拉使能控制： DISABLE：禁止 VLCD 下拉 ENABLE：使能 VLCD 下拉
输出参数	无
返回值	无

### 19.2.52 LCD\_LCDBSTCON\_VLCDPD\_Getable

函数名	LCD_LCDBSTCON_VLCDPD_Getable
函数原型	FunState LCD_LCDBSTCON_VLCDPD_Getable(void)
功能描述	获取 LCD VLCD 下拉状态
输入参数 1	无
输出参数	无
返回值	LCD VLCD 下拉使能控制：



DISABLE: 禁止 VLCD 下拉  
ENABLE: 使能 VLCD 下拉

### 19.2.53 LCD\_LCDBSTCON\_VLCDCFG\_Set

函数名	LCD_LCDBSTCON_VLCDCFG_Set
函数原型	void LCD_LCDBSTCON_VLCDCFG_Set(uint32_t SetValue)
功能描述	设置 LCD VLCD 输出电压
输入参数 1	LCD VLCD 输出电压: LCD_LCDBSTCON_VLCDCFG_2P5V: VLCD = 2.5V LCD_LCDBSTCON_VLCDCFG_3P0V: VLCD = 3.0V LCD_LCDBSTCON_VLCDCFG_3P5V: VLCD = 3.5V LCD_LCDBSTCON_VLCDCFG_4P0V: VLCD = 4.0V LCD_LCDBSTCON_VLCDCFG_4P5V: VLCD = 4.5V LCD_LCDBSTCON_VLCDCFG_5P0V: VLCD = 5.0V LCD_LCDBSTCON_VLCDCFG_5P5V: VLCD = 5.5V
输出参数	无
返回值	无

### 19.2.54 LCD\_LCDBSTCON\_VLCDCFG\_Get

函数名	LCD_LCDBSTCON_VLCDCFG_Get
函数原型	uint32_t LCD_LCDBSTCON_VLCDCFG_Get(void)
功能描述	获取 LCD VLCD 输出电压
输入参数 1	无
输出参数	无
返回值	LCD VLCD 输出电压: LCD_LCDBSTCON_VLCDCFG_2P5V: VLCD = 2.5V LCD_LCDBSTCON_VLCDCFG_3P0V: VLCD = 3.0V LCD_LCDBSTCON_VLCDCFG_3P5V: VLCD = 3.5V LCD_LCDBSTCON_VLCDCFG_4P0V: VLCD = 4.0V LCD_LCDBSTCON_VLCDCFG_4P5V: VLCD = 4.5V LCD_LCDBSTCON_VLCDCFG_5P0V: VLCD = 5.0V LCD_LCDBSTCON_VLCDCFG_5P5V: VLCD = 5.5V

### 19.2.55 LCD\_LCDBSTCON\_BOOST\_TEN\_Setable

函数名	LCD_LCDBSTCON_BOOST_TEN_Setable
-----	---------------------------------

函数原型	void LCD_LCDBSTCON_BOOST_TEN_Setable(FunState NewState)
功能描述	配置 LCD Booster 测试信号
输入参数 1	LCD Booster 测试信号： DISABLE: 禁止 Booster 测试信号 ENABLE: 使能 Booster 测试信号
输出参数	无
返回值	无

### 19.2.56 LCD\_LCDBSTCON\_BOOST\_TEN\_Getable

函数名	LCD_LCDBSTCON_BOOST_TEN_Getable
函数原型	FunState LCD_LCDBSTCON_BOOST_TEN_Getable(void)
功能描述	获取 LCD Booster 测试信号状态
输入参数 1	无
输出参数	无
返回值	LCD Booster 测试信号： DISABLE: 禁止 Booster 测试信号 ENABLE: 使能 Booster 测试信号

### 19.2.57 LCD\_LCDBSTCON\_BOOST\_EN\_Setable

函数名	LCD_LCDBSTCON_BOOST_EN_Setable
函数原型	void LCD_LCDBSTCON_BOOST_EN_Setable(FunState NewState)
功能描述	配置 LCD Booster 使能控制
输入参数 1	LCD Booster 使能控制： DISABLE: 禁止 Booster ENABLE: 使能 Booster
输出参数	无
返回值	无

### 19.2.58 LCD\_LCDBSTCON\_BOOST\_EN\_Getable

函数名	LCD_LCDBSTCON_BOOST_EN_Getable
函数原型	FunState LCD_LCDBSTCON_BOOST_EN_Getable(void)
功能描述	获取 LCD Booster 使能状态
输入参数 1	无
输出参数	无
返回值	LCD Booster 使能控制：



DISABLE: 禁止 Booster  
ENABLE: 使能 Booster

### 19.2.59 LCD\_COM\_EN\_Write

函数名	LCD_COM_EN_Write
函数原型	void LCD_COM_EN_Write(uint32_t SetValue)
功能描述	配置 LCD COM0~3 输出使能控制
输入参数 1	LCD COM0~3 输出使能控制: DISABLE: 禁止 COM 输出 ENABLE: 使能 COM 输出
输出参数	无
返回值	无

### 19.2.60 LCD\_COM\_EN\_Read

函数名	LCD_COM_EN_Read
函数原型	uint32_t LCD_COM_EN_Read(void)
功能描述	获取 LCD COM0~3 输出使能状态
输入参数 1	无
输出参数	无
返回值	LCD COM0~3 输出使能控制: DISABLE: 禁止 COM 输出 ENABLE: 使能 COM 输出

### 19.2.61 LCD\_SEG\_EN0\_Write

函数名	LCD_SEG_EN0_Write
函数原型	void LCD_SEG_EN0_Write(uint32_t SetValue)
功能描述	配置 LCD SEG0-31 输出使能控制
输入参数 1	LCD SEG0-31 输出使能控制: DISABLE: 禁止 SEG 输出 ENABLE: 使能 SEG 输出
输出参数	无
返回值	无



**19.2.62 LCD\_SEG\_EN0\_Read**

函数名	LCD_SEG_EN0_Read
函数原型	uint32_t LCD_SEG_EN0_Read(void)
功能描述	获取 LCD SEG0-31 输出使能状态
输入参数 1	无
输出参数	无
返回值	LCD SEG0-31 输出使能控制: DISABLE: 禁止 SEG 输出 ENABLE: 使能 SEG 输出

**19.2.63 LCD\_SEG\_EN1\_Write**

函数名	LCD_SEG_EN1_Write
函数原型	void LCD_SEG_EN1_Write(uint32_t SetValue)
功能描述	配置 LCD SEG32~43(COM4-7)输出使能控制
输入参数 1	LCD SEG32~43(COM4-7)输出使能控制: DISABLE: 禁止 SEG 输出 ENABLE: 使能 SEG 输出
输出参数	无
返回值	无

**19.2.64 LCD\_SEG\_EN1\_Read**

函数名	LCD_SEG_EN1_Read
函数原型	uint32_t LCD_SEG_EN1_Read(void)
功能描述	获取 LCD SEG32~43(COM4-7)输出使能状态
输入参数 1	无
输出参数	无
返回值	LCD SEG32~43(COM4-7)输出使能控制: DISABLE: 禁止 SEG 输出 ENABLE: 使能 SEG 输出

**19.2.65 LCD\_DISPDATAx\_Refresh**

函数名	LCD_DISPDATAx_Refresh
函数原型	void LCD_DISPDATAx_Refresh(uint32_t* DispBuf)
功能描述	设置 DISPDATAx 显示数据寄存器刷新





输入参数 1	显示 Buffer 数组:0-FFFFFFF
输出参数	无
返回值	无

### 19.2.66 LCD\_Boost\_Init

函数名	LCD_Boost_Init
函数原型	void LCD_Boost_Init(LCD_Boost_InitTypeDef* para)
功能描述	LCD Boost 功能初始化
输入参数 1	<p>LCD Boost 参数:</p> <p>LCD BOOST 环振频率:</p> <p>LCD_LCDBSTCON_OSC_CFG_GRADE0: 0 级 值越大频率越高, 驱动越强, 功耗越高  LCD_LCDBSTCON_OSC_CFG_GRADE1: 1 级 值越大频率越高, 驱动越强, 功耗越高  LCD_LCDBSTCON_OSC_CFG_GRADE2: 2 级 值越大频率越高, 驱动越强, 功耗越高  LCD_LCDBSTCON_OSC_CFG_GRADE3: 3 级 值越大频率越高, 驱动越强, 功耗越高  LCD_LCDBSTCON_OSC_CFG_GRADE4: 4 级 值越大频率越高, 驱动越强, 功耗越高  LCD_LCDBSTCON_OSC_CFG_GRADE5: 5 级 值越大频率越高, 驱动越强, 功耗越高  LCD_LCDBSTCON_OSC_CFG_GRADE6: 6 级 值越大频率越高, 驱动越强, 功耗越高  LCD_LCDBSTCON_OSC_CFG_GRADE7: 7 级 值越大频率越高, 驱动越强, 功耗越高  LCD_LCDBSTCON_OSC_CFG_GRADE8: 8 级 值越大频率越高, 驱动越强, 功耗越高</p> <p>VLCD 输出电压配置:</p> <p>LCD_LCDBSTCON_VLCDCFG_2P5V: VLCD = 2.5V  LCD_LCDBSTCON_VLCDCFG_3P0V: VLCD = 3.0V  LCD_LCDBSTCON_VLCDCFG_3P5V: VLCD = 3.5V  LCD_LCDBSTCON_VLCDCFG_4P0V: VLCD = 4.0V  LCD_LCDBSTCON_VLCDCFG_4P5V: VLCD = 4.5V  LCD_LCDBSTCON_VLCDCFG_5P0V: VLCD = 5.0V  LCD_LCDBSTCON_VLCDCFG_5P5V: VLCD = 5.5V</p> <p>基准源 Buffer 使能控制:</p> <p>DISABLE: 禁止基准源  ENABLE: 使能基准源</p> <p>基准源 Buffer Bypass 使能控制:</p> <p>DISABLE: 禁止基准源旁路  ENABLE: 使能基准源旁路</p> <p>VLCD 下拉使能控制:</p> <p>DISABLE: 禁止 VLCD 下拉  ENABLE: 使能 VLCD 下拉</p> <p>Booster 测试信号使能控制:</p> <p>DISABLE: 禁止 Booster 测试信号  ENABLE: 使能 Booster 测试信号</p> <p>Booster 使能控制:</p> <p>DISABLE: 禁止 Booster</p>

	ENABLE: 使能 Booster	
输出参数	无	
返回值	无	

### 19.2.67 LCD\_DISPCTRL\_ANTIPOLAR\_Setable

函数名	LCD_DISPCTRL_ANTIPOLAR_Setable
函数原型	void LCD_DISPCTRL_ANTIPOLAR_Setable(FunState NewState)
功能描述	设置 SEG/COM 在 LCD 关闭情况下接地控制, 防极化
输入参数 1	SEG/COM 在 LCD 关闭情况下接地控制: DISABLE: COM 和 SEG 在 LCD 关闭下浮空 ENABLE: COM 和 SEG 在 LCD 关闭下接地
输出参数	无
返回值	无

### 19.2.68 LCD\_DISPCTRL\_ANTIPOLAR\_Getable

函数名	LCD_DISPCTRL_ANTIPOLAR_Getable
函数原型	FunState LCD_DISPCTRL_ANTIPOLAR_Getable(void)
功能描述	获取 SEG/COM 在 LCD 关闭情况下接地状态
输入参数 1	无
输出参数	无
返回值	SEG/COM 在 LCD 关闭情况下接地控制: DISABLE: COM 和 SEG 在 LCD 关闭下浮空 ENABLE: COM 和 SEG 在 LCD 关闭下接地

## 20 UART

### 20.1 UART 寄存器结构

寄存器	描述
UARTIE	UART 中断使能寄存器
UARTIF	UART 中断标志寄存器

IRCON	红外调制设置寄存器
RXSTA0	接收状态控制寄存器
RXSTA1	
RXSTA2	
RXSTA3	
RXSTA4	
RXSTA5	
TXSTA0	发送状态控制寄存器
TXSTA1	
TXSTA2	
TXSTA3	
TXSTA4	
TXSTA5	
RXREG0	接收数据缓冲寄存器
RXREG1	
RXREG2	
RXREG3	
RXREG4	
RXREG5	
TXREG0	发送数据缓冲寄存器
TXREG1	
TXREG2	
TXREG3	
TXREG4	
TXREG5	
SPBRG0	波特率产生寄存器
SPBRG1	
SPBRG2	
SPBRG3	
SPBRG4	
SPBRG5	
TXBUFSTA0	发送 Buffer 状态控制寄存器
TXBUFSTA1	
TXBUFSTA2	
TXBUFSTA3	
TXBUFSTA4	
TXBUFSTA5	

RXBUFSTA0	接收 Buffer 状态控制寄存器
RXBUFSTA1	
RXBUFSTA2	
RXBUFSTA3	
RXBUFSTA4	
RXBUFSTA5	
RTXCON0	接收发送取反控制寄存器
RTXCON1	
RTXCON2	
RTXCON3	
RTXCON4	
RTXCON5	

## 20.2 UART 库函数

序号	函数名	描述
1	UART_Init	UART 完整参数初始化
2	UART_SInit	UART 简单参数初始化
3	UART_All_Deinit	UART 全部初始化为缺省值
4	UART_Deinit	UART 的 UARTIE 和 UARTIF 初始化为缺省值
5	UARTx_Deinit	UARTx 初始化为缺省值
6	UARTx_TXREG_Write	写入发送数据缓冲寄存器的值
7	UARTx_RXREG_Read	读取接收数据缓冲寄存器
8	UARTx_SPBRG_Write	写入波特率
9	UARTx_SPBRG_Read	读取波特率
10	UART_BaudREGCalc	计算 uart 波特率
11	UARTx_TXSTA_TXEN_Setable	设置发送模块使能位
12	UARTx_TXSTA_TXEN_Getable	读取发送模块使能位设置
13	UARTx_RXSTA_RXEN_Setable	设置接收模块使能位
14	UARTx_RXSTA_RXEN_Getable	读取接收模块使能位设置
15	UARTx_RXSTA_PDSEL_Set	设置模式选择位
16	UARTx_RXSTA_PDSEL_Get	读取模式选择位的设置
17	UARTx_RTXCON_RTX7EN_Setable	设置收发 7bit 数据使能
18	UARTx_RTXCON_RTX7EN_Getable	读取收发 7bit 数据使能设置
19	UARTx_TXSTA_TX9D_Set	设置发送数据的第 9 位



20	UARTx_RXSTA_RX9D_Chk	读取接收数据的第 9 位
21	UARTx_TXSTA_STOPSEL_Set	设置停止位选择位
22	UARTx_TXSTA_STOPSEL_Get	读取停止位选择位设置
23	UARTx_TXSTA_TXIS_Set	设置发送中断选择位
24	UARTx_TXSTA_TXIS_Get	读取发送中断选择位设置
25	UARTx_TXBUFSTA_TX_INTSEL_Set	设置发送中断选择位
26	UARTx_TXBUFSTA_TX_INTSEL_Get	读取发送中断选择位设置
27	UARTx_TXBUFSTA_TXFF_Chk	读取 TXBUF 状态位
28	UARTx_RXBUFSTA_RXFF_Chk	读取 RXBUF 状态位
29	UART_UARTIE_RxTxIE_SetableEx	设置 uart 接收 发送 中断使能
30	UART_UARTIE_RxTxIE_GetableEx	读取 uart 接收 发送 中断使能状态
31	UART_UARTIF_RxTxIF_ClrEx	清 uart 接收 发送 中断标志
32	UART_UARTIF_RxTxIF_ChkEx	读取 uart 接收 发送 中断标志
33	UARTx_RXSTA_ERRIE_Setable	设置错误中断允许位
34	UARTx_RXSTA_ERRIE_Getable	读取错误中断允许位设置
35	UARTx_RXSTA_PERR_Clr	清奇偶校验错标志位
36	UARTx_RXSTA_PERR_Chk	读取奇偶校验错标志
37	UARTx_RXSTA_FERR_Clr	清帧格式错标志位
38	UARTx_RXSTA_FERR_Chk	读取帧格式错标志位
39	UARTx_RXSTA_OERR_Clr	清溢出错标志位
40	UARTx_RXSTA_OERR_Chk	读取溢出错标志位
41	UARTx_RTXCON_TXDFLAG_Setable	设置发送数据取反控制位
42	UARTx_RTXCON_TXDFLAG_Getable	读取发送数据取反控制位的设置
43	UARTx_RTXCON_RXDFLAG_Setable	设置接收数据取反控制位
44	UARTx_RTXCON_RXDFLAG_Getable	读取接收数据取反控制位的设置
45	UARTx_TXSTA_IREN_Setable	设置发送红外调制使能位
46	UARTx_TXSTA_IREN_Getable	读取发送红外调制使能位设置
47	UART_IRModulation_Init	设置红外调制频率占空比
48	UART_IRCON_TH_Set	设置红外占空比调制参数
49	UART_IRCON_TH_Get	读取红外占空比调制参数
50	UART_IRCON_TZBRG_Set	设置红外调制频率
51	UART_IRCON_TZBRG_Get	读取红外调制频率的设置
52	UART_IRCON_IRFLAG_Setable	设置红外调制发送数据极性选择
53	UART_IRCON_IRFLAG_Getable	读取红外调制发送数据极性选择

## 20.2.1 UART\_Init

函数名	UART_Init
函数原型	void UART_Init(UARTx_Type* UARTx, UART_InitTypeDef* para)
功能描述	UART 完整参数初始化
输入参数 1	UART0~UART5
输入参数 2	<div>UART 完整参数：</div> <div><div>接收中断控制： DISABLE：禁止接收中断 ENABLE：使能接收中断</div><div>发送中断控制： DISABLE：禁止发送中断 ENABLE：使能发送中断</div><div>波特率分频值：0-FFFF</div><div>串口模式选择： 00：8 位数据，无奇偶校验 01：8 位数据，偶校验 10：8 位数据，奇校验 11：9 位数据，无奇偶校验</div><div>错误中断使能控制： DISABLE：禁止错误中断 ENABLE：使能错误中断</div><div>接收模块使能控制： DISABLE：禁止接收模块 ENABLE：使能接收模块</div><div>TX_INTSEL=0 时，发送中断选择位： 0：发送缓冲器空产生中断 1：移位寄存器空产生中断</div><div>发送模块使能控制： DISABLE：禁止发送模块 ENABLE：使能发送模块</div><div>发送红外调制使能： DISABLE：禁止发送红外调制 ENABLE：使能发送红外调制</div><div>发送中断选择： 00：由 TXIS 决定发送中断在发送缓冲器空或者移位寄存器空时产生 01：TXBUF 空且移位寄存器空产生中断 10：TXBUF 空产生中断 11：不产生中断</div><div>收发 7bit 数据使能： DISABLE：禁止收发 7bit 数据 1：允许收发 7bit 数据</div><div>接收数据取反控制：</div></div>

	0: 接收数据不取反 1: 接收数据取反 发送数据取反控制: 0: 发送数据不取反 1: 发送数据取反	
输出参数	无	
返回值	无	

### 20.2.2 UART\_SInit

函数名	UART_SInit
函数原型	void UART_SInit(UARTx_Type* UARTx, UART_SInitTypeDef* para, uint32_t APBCLK)
功能描述	UART 简单参数初始化
输入参数 1	UART0~UART5
输入参数 2	<div>UART 简单参数:</div> <div><div>波特率</div><div>数据位数: 0: 7 位数据不支奇偶校验位 1: 8 位数据 2: 9 位数据不支持奇偶校验位</div><div>校验位: 0: 无校验位 1: 偶校验 2: 奇校验</div><div>停止位: 0: 1 位停止位 1: 2 位停止位</div></div>
输入参数 3	APB 时钟频率
输出参数	无
返回值	无

### 20.2.3 UART\_All\_Deinit

函数名	UART_All_Deinit	
函数原型	void UART_All_Deinit(void)	
功能描述	UART 全部初始化为缺省值	
输入参数 1	无	
输出参数	无	
返回值	无	



## 20.2.4 UART\_Deinit

函数名	UART_Deinit
函数原型	void UART_Deinit(void)
功能描述	UART 的 UARTIE 和 UARTIF 初始化为缺省值
输入参数 1	无
输出参数	无
返回值	无

## 20.2.5 UARTx\_Deinit

函数名	UARTx_Deinit
函数原型	void UARTx_Deinit(UARTx_Type* UARTx)
功能描述	UARTx 初始化为缺省值
输入参数 1	UART0~UART5
输出参数	无
返回值	无

## 20.2.6 UARTx\_TXREG\_Write

函数名	UARTx_TXREG_Write
函数原型	void UARTx_TXREG_Write(UARTx_Type* UARTx, uint32_t SetValue)
功能描述	写入发送数据缓冲寄存器的值
输入参数 1	UART0~UART5
输入参数 2	要发送的数值: 0-FF
输出参数	无
返回值	无

## 20.2.7 UARTx\_RXREG\_Read

函数名	UARTx_RXREG_Read
函数原型	uint32_t UARTx_RXREG_Read(UARTx_Type* UARTx)
功能描述	读取接收数据缓冲寄存器
输入参数 1	UART0~UART5
输出参数	无
返回值	接收数据缓冲寄存器的数值:0-FF



## 20.2.8 UARTx\_SPBRG\_Write

函数名	UARTx_SPBRG_Write
函数原型	void UARTx_SPBRG_Write(UARTx_Type* UARTx, uint32_t SetValue)
功能描述	写入波特率分频值
输入参数 1	UART0~UART5
输入参数 2	波特率分频值:0-FFFF
输出参数	无
返回值	无

## 20.2.9 UARTx\_SPBRG\_Read

函数名	UARTx_SPBRG_Read
函数原型	uint32_t UARTx_SPBRG_Read(UARTx_Type* UARTx)
功能描述	读取波特率分频值
输入参数 1	UART0~UART5
输出参数	无
返回值	波特率分频值： 0-FFFF

## 20.2.10 UART\_BaudREGCalc

函数名	UART_BaudREGCalc
函数原型	uint32_t UART_BaudREGCalc(uint32_t BaudRate, uint32_t APBCLK)
功能描述	计算 uart 波特率寄存器值
输入参数 1	uart 波特率： 0-APBCLK
输入参数 2	APB 时钟频率
输出参数	无
返回值	uart 波特率寄存器值

## 20.2.11 UARTx\_TXSTA\_TXEN\_Setable

函数名	UARTx_TXSTA_TXEN_Setable
函数原型	void UARTx_TXSTA_TXEN_Setable(UARTx_Type* UARTx, FunState NewState)
功能描述	设置发送模块使能位
输入参数 1	UART0~UART5
输入参数 2	发送模块使能控制： DISABLE：禁止发送模块 ENABLE：使能发送模块

输出参数	无
返回值	无

### 20.2.12 UARTx\_TXSTA\_TXEN\_Getable

函数名	UARTx_TXSTA_TXEN_Getable
函数原型	FunState UARTx_TXSTA_TXEN_Getable(UARTx_Type* UARTx)
功能描述	读取发送模块使能位设置
输入参数 1	UART0~UART5
输出参数	无
返回值	发送模块使能控制： DISABLE：禁止发送模块 ENABLE：使能发送模块

### 20.2.13 UARTx\_RXSTA\_RXEN\_Setable

函数名	UARTx_RXSTA_RXEN_Setable
函数原型	void UARTx_RXSTA_RXEN_Setable(UARTx_Type* UARTx, FunState NewState)
功能描述	设置接收模块使能位
输入参数 1	UART0~UART5
输入参数 2	接收模块使能控制： DISABLE：禁止接收模块 ENABLE：使能接收模块
输出参数	无
返回值	无

### 20.2.14 UARTx\_RXSTA\_RXEN\_Getable

函数名	UARTx_RXSTA_RXEN_Getable
函数原型	FunState UARTx_RXSTA_RXEN_Getable(UARTx_Type* UARTx)
功能描述	读取接收模块使能位设置
输入参数 1	UART0~UART5
输出参数	无
返回值	接收模块使能控制： DISABLE：禁止接收模块 ENABLE：使能接收模块

**20.2.15 UARTx\_RXSTA\_PDSEL\_Set**

函数名	UARTx_RXSTA_PDSEL_Set
函数原型	void UARTx_RXSTA_PDSEL_Set(UARTx_Type* UARTx, uint32_t SetValue)
功能描述	设置模式选择
输入参数 1	UART0~UART5
输入参数 2	串口模式选择: 00: 8 位数据, 无奇偶校验 01: 8 位数据, 偶校验 10: 8 位数据, 奇校验 11: 9 位数据, 无奇偶校验
输出参数	无
返回值	无

**20.2.16 UARTx\_RXSTA\_PDSEL\_Get**

函数名	UARTx_RXSTA_PDSEL_Get
函数原型	uint32_t UARTx_RXSTA_PDSEL_Get(UARTx_Type* UARTx)
功能描述	读取模式选择的设置
输入参数 1	UART0~UART5
输出参数	无
返回值	串口模式选择: 00: 8 位数据, 无奇偶校验 01: 8 位数据, 偶校验 10: 8 位数据, 奇校验 11: 9 位数据, 无奇偶校验

**20.2.17 UARTx\_RTXCON\_RTX7EN\_Setable**

函数名	UARTx_RTXCON_RTX7EN_Setable
函数原型	void UARTx_RTXCON_RTX7EN_Setable(UARTx_Type* UARTx, FunState NewState)
功能描述	设置收发 7bit 数据使能
输入参数 1	UART0~UART5
输入参数 2	收发 7bit 数据使能: 0: 正常收发 1: 收发 7 位数据帧, 格式为 7 位数据位+STOP
输出参数	无
返回值	无

### 20.2.18 UARTx\_RTXCON\_RTX7EN\_Getable

函数名	UARTx_RTXCON_RTX7EN_Getable
函数原型	FunState UARTx_RTXCON_RTX7EN_Getable(UARTx_Type* UARTx)
功能描述	读取收发 7bit 数据使能设置
输入参数 1	UART0~UART5
输出参数	无
返回值	收发 7bit 数据使能： 0：正常收发 1：收发 7 位数据帧，格式为 7 位数据位+STOP

### 20.2.19 UARTx\_TXSTA\_TX9D\_Set

函数名	UARTx_TXSTA_TX9D_Set
函数原型	void UARTx_TXSTA_TX9D_Set(UARTx_Type* UARTx, uint32_t SetValue)
功能描述	设置发送数据的第 9 位
输入参数 1	UART0~UART5
输入参数 2	发送数据的第 9 位
输出参数	无
返回值	无

### 20.2.20 UARTx\_RXSTA\_RX9D\_Chk

函数名	UARTx_RXSTA_RX9D_Chk
函数原型	FlagStatus UARTx_RXSTA_RX9D_Chk(UARTx_Type* UARTx)
功能描述	读取接收数据的第 9 位
输入参数 1	UART0~UART5
输出参数	无
返回值	接收数据的第 9 位

### 20.2.21 UARTx\_TXSTA\_STOPSEL\_Set

函数名	UARTx_TXSTA_STOPSEL_Set
函数原型	void UARTx_TXSTA_STOPSEL_Set(UARTx_Type* UARTx, uint32_t SetValue)
功能描述	设置停止位选择位
输入参数 1	UART0~UART5
输入参数 2	停止位选择：

	0: 停止位为 1 位 1: 停止位为 2 位
输出参数	无
返回值	无

### 20.2.22 UARTx\_TXSTA\_STOPSEL\_Get

函数名	UARTx_TXSTA_STOPSEL_Get
函数原型	uint32_t UARTx_TXSTA_STOPSEL_Get(UARTx_Type* UARTx)
功能描述	读取停止位选择位设置
输入参数 1	UART0~UART5
输出参数	无
返回值	停止位选择: 0: 停止位为 1 位 1: 停止位为 2 位

### 20.2.23 UARTx\_TXSTA\_TXIS\_Set

函数名	UARTx_TXSTA_TXIS_Set
函数原型	void UARTx_TXSTA_TXIS_Set(UARTx_Type* UARTx, uint32_t SetValue)
功能描述	发送中断选择: 0: 发送缓冲器空产生中断 1: 移位寄存器空产生中断
输入参数 1	UART0~UART5
输出参数	无
返回值	无

### 20.2.24 UARTx\_TXSTA\_TXIS\_Get

函数名	UARTx_TXSTA_TXIS_Get
函数原型	uint32_t UARTx_TXSTA_TXIS_Get(UARTx_Type* UARTx)
功能描述	读取发送中断选择位设置
输入参数 1	UART0~UART5
输出参数	无
返回值	发送中断选择: 0: 发送缓冲器空产生中断 1: 移位寄存器空产生中断

**20.2.25 UARTx\_TXBUFSTA\_TX\_INTSEL\_Set**

函数名	UARTx_TXBUFSTA_TX_INTSEL_Set
函数原型	void UARTx_TXBUFSTA_TX_INTSEL_Set(UARTx_Type* UARTx, uint32_t SetValue)
功能描述	设置发送中断选择位
输入参数 1	UART0~UART5
输入参数 2	发送中断选择: 00: 由 TXIS 决定发送中断在发送缓冲器空或者移位寄存器空时产生 01: TXBUF 空且移位寄存器空产生中断 10: TXBUF 空产生中断 11: 不产生中断
输出参数	无
返回值	无

**20.2.26 UARTx\_TXBUFSTA\_TX\_INTSEL\_Get**

函数名	UARTx_TXBUFSTA_TX_INTSEL_Get
函数原型	uint32_t UARTx_TXBUFSTA_TX_INTSEL_Get(UARTx_Type* UARTx)
功能描述	读取发送中断选择位设置
输入参数 1	UART0~UART5
输出参数	无
返回值	发送中断选择: 00: 由 TXIS 决定发送中断在发送缓冲器空或者移位寄存器空时产生 01: TXBUF 空且移位寄存器空产生中断 10: TXBUF 空产生中断 11: 不产生中断

**20.2.27 UARTx\_TXBUFSTA\_TXFF\_Chk**

函数名	UARTx_TXBUFSTA_TXFF_Chk
函数原型	FlagStatus UARTx_TXBUFSTA_TXFF_Chk(UARTx_Type* UARTx)
功能描述	读取 TXBUF 状态位
输入参数 1	UART0~UART5
输出参数	无
返回值	TXBUF 状态位: 0: TXBUF 空 1: TXBUF 中有数据

**20.2.28 UARTx\_RXBUFSTA\_RXFF\_Chk**

函数名	UARTx_RXBUFSTA_RXFF_Chk
函数原型	FlagStatus UARTx_RXBUFSTA_RXFF_Chk(UARTx_Type* UARTx)
功能描述	读取 RXBUF 状态位
输入参数 1	UART0~UART5
输出参数	无
返回值	RXBUF 状态位: 0: RXBUF 空 1: RXBUF 中有数据

**20.2.29 UART\_UARTIE\_RxTxIE\_SetableEx**

函数名	UART_UARTIE_RxTxIE_SetableEx
函数原型	void UART_UARTIE_RxTxIE_SetableEx(UARTx_Type* UARTx, UART_IntTypeDef IntType, FunState NewState)
功能描述	设置 uart 收发中断使能
输入参数 1	UART0~UART5
输入参数 2	收发中断选择: 0: 接收中断 1: 发送中断
输入参数 3	收发中断使能: DISABLE: 禁止收发中断 ENABLE: 使能收发中断
输出参数	无
返回值	无

**20.2.30 UART\_UARTIE\_RxTxIE\_GetableEx**

函数名	UART_UARTIE_RxTxIE_GetableEx
函数原型	FunState UART_UARTIE_RxTxIE_GetableEx(UARTx_Type* UARTx, UART_IntTypeDef IntType)
功能描述	读取 uart 收发中断使能状态
输入参数 1	UART0~UART5
输入参数 2	收发中断选择: 0: 接收中断 1: 发送中断
输出参数	无

返回值	收发中断使能： DISABLE：禁止收发中断 ENABLE：使能收发中断
-----	--------------------------------------------

### 20.2.31 UART\_UARTIF\_RxTxIF\_ClrEx

函数名	UART_UARTIF_RxTxIF_ClrEx
函数原型	void UART_UARTIF_RxTxIF_ClrEx(UARTx_Type* UARTx)
功能描述	清除 uart 收发中断标志
输入参数 1	UART0~UART5
输出参数	无
返回值	无

### 20.2.32 UART\_UARTIF\_RxTxIF\_ChkEx

函数名	UART_UARTIF_RxTxIF_ChkEx
函数原型	FlagStatus UART_UARTIF_RxTxIF_ChkEx(UARTx_Type* UARTx, UART_IntTypeDef IntType)
功能描述	读取 uart 收发中断标志
输入参数 1	UART0~UART5
输入参数 2	收发中断选择： 0：接收中断 1：发送中断
输出参数	无
返回值	收发中断标志： 0：未发生收发中断 1：发生收发中断

### 20.2.33 UARTx\_RXSTA\_ERRIE\_Setable

函数名	UARTx_RXSTA_ERRIE_Setable
函数原型	void UARTx_RXSTA_ERRIE_Setable(UARTx_Type* UARTx, FunState NewState)
功能描述	设置错误中断允许位
输入参数 1	UART0~UART5
输入参数 2	错误中断使能控制： DISABLE：禁止错误中断 ENABLE：使能错误中断
输出参数	无



返回值	无
-----	---

### 20.2.34 UARTx\_RXSTA\_ERRIE\_Getable

函数名	UARTx_RXSTA_ERRIE_Getable
函数原型	FunState UARTx_RXSTA_ERRIE_Getable(UARTx_Type* UARTx)
功能描述	读取错误中断允许位设置
输入参数 1	UART0~UART5
输出参数	无
返回值	错误中断使能控制： DISABLE：禁止错误中断 ENABLE：使能错误中断

### 20.2.35 UARTx\_RXSTA\_PERR\_Clr

函数名	UARTx_RXSTA_PERR_Clr
函数原型	void UARTx_RXSTA_PERR_Clr(UARTx_Type* UARTx)
功能描述	清除奇偶校验错标志位
输入参数 1	UART0~UART5
输出参数	无
返回值	无

### 20.2.36 UARTx\_RXSTA\_PERR\_Chk

函数名	UARTx_RXSTA_PERR_Chk
函数原型	FlagStatus UARTx_RXSTA_PERR_Chk(UARTx_Type* UARTx)
功能描述	读取奇偶校验错标志
输入参数 1	UART0~UART5
输出参数	无
返回值	0：无检验错 1：校验错

### 20.2.37 UARTx\_RXSTA\_FERR\_Clr

函数名	UARTx_RXSTA_FERR_Clr
函数原型	void UARTx_RXSTA_FERR_Clr(UARTx_Type* UARTx)
功能描述	清帧格式错标志位
输入参数 1	UART0~UART5



输出参数	无
返回值	无

### 20.2.38 UARTx\_RXSTA\_FERR\_Chk

函数名	UARTx_RXSTA_FERR_Chk
函数原型	FlagStatus UARTx_RXSTA_FERR_Chk(UARTx_Type* UARTx)
功能描述	读取帧格式错标志位
输入参数 1	UART0~UART5
输出参数	无
返回值	0: 无数据帧格式错 1: 数据帧格式错

### 20.2.39 UARTx\_RXSTA\_OERR\_Clr

函数名	UARTx_RXSTA_OERR_Clr
函数原型	void UARTx_RXSTA_OERR_Clr(UARTx_Type* UARTx)
功能描述	清溢出错标志位
输入参数 1	UART0~UART5
输出参数	无
返回值	无

### 20.2.40 UARTx\_RXSTA\_OERR\_Chk

函数名	UARTx_RXSTA_OERR_Chk
函数原型	FlagStatus UARTx_RXSTA_OERR_Chk(UARTx_Type* UARTx)
功能描述	读取溢出错标志位
输入参数 1	UART0~UART5
输出参数	0: 无溢出错 1: 溢出错
返回值	无

### 20.2.41 UARTx\_RTXCON\_TXDFLAG\_Setable

函数名	UARTx_RTXCON_TXDFLAG_Setable
函数原型	void UARTx_RTXCON_TXDFLAG_Setable(UARTx_Type* UARTx, FunState NewState)
功能描述	设置发送数据取反控制位

输入参数 1	UART0~UART5
输入参数 2	发送数据取反控制： 0：发送数据不取反 1：发送数据取反
输出参数	无
返回值	无

#### 20.2.42 UARTx\_RTXCON\_TXDFLAG\_Getable

函数名	UARTx_RTXCON_TXDFLAG_Getable
函数原型	FunState UARTx_RTXCON_TXDFLAG_Getable(UARTx_Type* UARTx)
功能描述	读取发送数据取反控制位的设置
输入参数 1	UART0~UART5
输出参数	无
返回值	发送数据取反控制： 0：发送数据不取反 1：发送数据取反

#### 20.2.43 UARTx\_RTXCON\_RXDFLAG\_Setable

函数名	UARTx_RTXCON_RXDFLAG_Setable
函数原型	void UARTx_RTXCON_RXDFLAG_Setable(UARTx_Type* UARTx, FunState NewState)
功能描述	设置接收数据取反控制位
输入参数 1	UART0~UART5
输入参数 2	发送数据取反控制： 0：发送数据不取反 1：发送数据取反
输出参数	无
返回值	无

#### 20.2.44 UARTx\_RTXCON\_RXDFLAG\_Getable

函数名	UARTx_RTXCON_RXDFLAG_Getable
函数原型	FunState UARTx_RTXCON_RXDFLAG_Getable(UARTx_Type* UARTx)
功能描述	读取接收数据取反控制位的设置
输入参数 1	UART0~UART5
输出参数	无

返回值	接收数据取反控制： 0：接收数据不取反 1：接收数据取反
-----	------------------------------------

#### 20.2.45 UARTx\_TXSTA\_IREN\_Setable

函数名	UARTx_TXSTA_IREN_Setable
函数原型	void UARTx_TXSTA_IREN_Setable(UARTx_Type* UARTx, FunState NewState)
功能描述	设置发送红外调制使能位
输入参数 1	UART0~UART5
输入参数 2	发送红外调制使能： DISABLE：禁止发送红外调制 ENABLE：使能发送红外调制
输出参数	无
返回值	无

#### 20.2.46 UARTx\_TXSTA\_IREN\_Getable

函数名	UARTx_TXSTA_IREN_Getable
函数原型	FunState UARTx_TXSTA_IREN_Getable(UARTx_Type* UARTx)
功能描述	读取发送红外调制使能位设置
输入参数 1	UART0~UART5
输出参数	无
返回值	发送红外调制使能： DISABLE：禁止发送红外调制 ENABLE：使能发送红外调制

#### 20.2.47 UART\_IRModulation\_Init

函数名	UART_IRModulation_Init
函数原型	void UART_IRModulation_Init( uint32_t ModuFreq, uint8_t ModuDutyCycle, uint32_t APBClk )
功能描述	设置红外调制频率占空比
输入参数 1	红外频率
输入参数 2	占空比
输入参数 3	APB 时钟频率
输出参数	无
返回值	无

**20.2.48 UART\_IRCON\_TH\_Set**

函数名	UART_IRCON_TH_Set
函数原型	void UART_IRCON_TH_Set(uint32_t SetValue)
功能描述	设置红外占空比调制参数
输入参数 1	红外占空比调制参数
输出参数	无
返回值	无

**20.2.49 UART\_IRCON\_TH\_Get**

函数名	UART_IRCON_TH_Get
函数原型	uint32_t UART_IRCON_TH_Get(void)
功能描述	读取红外占空比调制参数
输入参数 1	无
输出参数	无
返回值	红外占空比调制参数

**20.2.50 UART\_IRCON\_TZBRG\_Set**

函数名	UART_IRCON_TZBRG_Set
函数原型	void UART_IRCON_TZBRG_Set(uint32_t SetValue)
功能描述	设置红外调制频率
输入参数 1	红外调制频率
输出参数	无
返回值	无

**20.2.51 UART\_IRCON\_TZBRG\_Get**

函数名	UART_IRCON_TZBRG_Get
函数原型	uint32_t UART_IRCON_TZBRG_Get(void)
功能描述	读取红外调制频率的设置
输入参数 1	无
输出参数	无
返回值	红外调制频率

**20.2.52 UART\_IRCON\_IRFLAG\_Setable**

函数名	UART_IRCON_IRFLAG_Setable
函数原型	void UART_IRCON_IRFLAG_Setable(FunState NewState)
功能描述	设置红外调制发送数据极性选择
输入参数 1	DISABLE: 正极性输出适合 PNP 管驱动 ENABLE: 负极性输出适合 NPN 管驱动
输出参数	无
返回值	无

**20.2.53 UART\_IRCON\_IRFLAG\_Getable**

函数名	UART_IRCON_IRFLAG_Getable
函数原型	FunState UART_IRCON_IRFLAG_Getable(void)
功能描述	读取红外调制发送数据极性选择
输入参数 1	无
输出参数	无
返回值	DISABLE: 正极性输出适合 PNP 管驱动 ENABLE: 负极性输出适合 NPN 管驱动

## 21 I2C

### 21.1 I2C 寄存器结构

寄存器	描述
I2CCTRL	I2C 控制寄存器
I2CSTA	I2C 状态寄存器
I2CBRG	I2C 波特率设置寄存器
I2CBUF	I2C 收发缓冲寄存器
I2CIR	I2C 中断寄存器
I2CFSM	I2C 状态机查询寄存器
I2CERR	I2C 错误标志寄存器

### 21.2 I2C 库函数

序号	函数名	描述
1	I2C_Deinit	I2C 初始化为缺省值
2	I2C_I2CCTRL_I2CEN_Setable	设置 I2C 模块使能控制
3	I2C_I2CCTRL_I2CEN_Getable	读取 I2C 模块使能控制位设置
4	I2C_I2CBUF_Write	写收发缓冲寄存器
5	I2C_I2CBUF_Read	读收发缓冲寄存器
6	I2C_I2CCTRL_SCLHL_Setable	I2C 模块禁止时, 设置 SCL 固定为高低电平控制位
7	I2C_I2CCTRL_SCLHL_Getable	读取 SCL 固定为高低电平控制位的设置
8	I2C_I2CCTRL_SDAHL_Setable	I2C 模块禁止时, 设置 SDA 固定为高低电平控制位
9	I2C_I2CCTRL_SDAHL_Getable	读取 SDA 固定为高低电平控制位的设置
10	I2C_I2CCTRL_ACKEN_Setable	主控接收模式下, 设置主机给从机回应使能位
11	I2C_I2CCTRL_ACKEN_Getable	主控接收模式下, 读取主机给从机回应使能位
12	I2C_I2CCTRL_RCEN_Setable	主控接收模式下, 设置接收使能位
13	I2C_I2CCTRL_RCEN_Getable	主控接收模式下, 读取接收使能位

14	I2C_I2CCTRL_RSEN_Setable	设置 Repeated START 时序产生使能控制位
15	I2C_I2CCTRL_PEN_Setable	设置 STOP 时序产生使能控制位
16	I2C_I2CCTRL_SEN_Setable	设置 START 时序产生使能控制位
17	I2C_I2CBRG_Write	设置波特率
18	I2C_I2CBRG_Read	读取波特率
19	I2C_BaudREG_Calc	I2C 波特率寄存器值计算
20	I2C_I2CSTA_RW_Chk	读取 I2C 接口状态
21	I2C_I2CSTA_BF_Chk	读取 缓冲器满状态位
22	I2C_I2CSTA_S_Chk	读取 START 标志位
23	I2C_I2CSTA_P_Chk	读取 STOP 标志位
24	I2C_I2CSTA_WCOL_Clr	清除写冲突检测位
25	I2C_I2CSTA_WCOL_Chk	读取写冲突检测位
26	I2C_I2CSTA_ACKSTA_Clr	清除来自从机的回应信号
27	I2C_I2CSTA_ACKSTA_Chk	读取 来自从机的回应信号
28	I2C_I2CSTA_ACKDT_Set	设置主机回应信号的状态
29	I2C_I2CSTA_ACKDT_Get	读取主机回应信号的状态设置
30	I2C_I2CIR_I2CIE_Setable	设置 I2C 中断使能控制位
31	I2C_I2CIR_I2CIE_Getable	读取 I2C 中断使能控制位设置
32	I2C_I2CIR_I2CIF_Clr	清除 I2C 中断标志位
33	I2C_I2CIR_I2CIF_Chk	读取 I2C 中断标志位
34	I2C_I2CFSM_I2CFSM_Get	读取 I2C 主状态机编码
35	I2C_I2CERR_ERRIE_Setable	设置错误标志中断使能
36	I2C_I2CERR_ERRIE_Getable	读取错误标志中断使能设置
37	I2C_I2CERR_OIERR_Clr	清除 OP_IDLE 状态下错误标志位
38	I2C_I2CERR_OIERR_Chk	读取 OP_IDLE 状态下错误标志位
39	I2C_I2CERR_SDERR_Clr	清除 START_DONE 状态下错误标志位
40	I2C_I2CERR_SDERR_Chk	读取 START_DONE 状态下错误标志位
41	I2C_I2CERR_IERR_Clr	清除 IDLE 状态下错误标志位
42	I2C_I2CERR_IERR_Chk	读取 IDLE 状态下错误标志位

### 21.2.1 I2C\_Deinit

函数名	I2C_Deinit
函数原型	void I2C_Deinit(void)



功能描述	I2C 初始化为缺省值
输入参数 1	无
输出参数	无
返回值	无

### 21.2.2 I2C\_I2CCTRL\_I2CEN\_Setable

函数名	I2C_I2CCTRL_I2CEN_Setable
函数原型	void I2C_I2CCTRL_I2CEN_Setable(FunState NewState)
功能描述	设置 I2C 模块使能控制
输入参数 1	I2C 模块使能控制： DISABLE: 禁止 I2C 模块 ENABLE: 使能 I2C 模块
输出参数	无
返回值	无

### 21.2.3 I2C\_I2CCTRL\_I2CEN\_Getable

函数名	I2C_I2CCTRL_I2CEN_Getable
函数原型	FunState I2C_I2CCTRL_I2CEN_Getable(void)
功能描述	读取 I2C 模块使能控制位设置
输入参数 1	无
输出参数	无
返回值	I2C 模块使能控制： DISABLE: 禁止 I2C 模块 ENABLE: 使能 I2C 模块

### 21.2.4 I2C\_I2CBUF\_Write

函数名	I2C_I2CBUF_Write
函数原型	void I2C_I2CBUF_Write(uint32_t SetValue)
功能描述	写收发缓冲寄存器
输入参数 1	写入收发缓冲寄存器的值
输出参数	无
返回值	无

### 21.2.5 I2C\_I2CBUF\_Read

函数名	I2C_I2CBUF_Read
函数原型	uint32_t I2C_I2CBUF_Read(void)
功能描述	读收发缓冲寄存器
输入参数 1	无
输出参数	无
返回值	收发缓冲寄存器的值

### 21.2.6 I2C\_I2CCTRL\_SCLHL\_Setable

函数名	I2C_I2CCTRL_SCLHL_Setable
函数原型	void I2C_I2CCTRL_SCLHL_Setable(FunState NewState)
功能描述	I2C 模块禁止时, 设置 SCL 固定为高低电平控制位
输入参数 1	设置 SCL 固定的高低电平控制位: 0: SCL 固定为低电平 1: SCL 固定为高电平
输出参数	无
返回值	无

### 21.2.7 I2C\_I2CCTRL\_SCLHL\_Getable

函数名	I2C_I2CCTRL_SCLHL_Getable
函数原型	FunState I2C_I2CCTRL_SCLHL_Getable(void)
功能描述	读取 SCL 固定为高低电平控制位的设置
输入参数 1	无
输出参数	无
返回值	设置 SCL 固定的高低电平控制位: 0: SCL 固定为低电平 1: SCL 固定为高电平

### 21.2.8 I2C\_I2CCTRL\_SDAHL\_Setable

函数名	I2C_I2CCTRL_SDAHL_Setable
函数原型	void I2C_I2CCTRL_SDAHL_Setable(FunState NewState)
功能描述	I2C 模块禁止时, 设置 SDA 固定为高低电平控制位
输入参数 1	设置 SDA 固定的高低电平控制位: 0: SDA 固定为低电平

	1: SDA 固定为高电平
输出参数	无
返回值	无

### 21.2.9 I2C\_I2CCTRL\_SDAHL\_Getable

函数名	I2C_I2CCTRL_SDAHL_Getable
函数原型	FunState I2C_I2CCTRL_SDAHL_Getable(void)
功能描述	读取 SDA 固定为高低电平控制位的设置
输入参数 1	无
输出参数	无
返回值	设置 SDA 固定的高低电平控制位： 0: SDA 固定为低电平 1: SDA 固定为高电平

### 21.2.10 I2C\_I2CCTRL\_ACKEN\_Setable

函数名	I2C_I2CCTRL_ACKEN_Setable
函数原型	void I2C_I2CCTRL_ACKEN_Setable(FunState NewState)
功能描述	主控接收模式下，设置主机给从机回应使能位
输入参数 1	主机给从机回应使能位： 0: 主机不响应从机 1: 主机发送回应 ACKDT 给从机
输出参数	无
返回值	无

### 21.2.11 I2C\_I2CCTRL\_ACKEN\_Getable

函数名	I2C_I2CCTRL_ACKEN_Getable
函数原型	FunState I2C_I2CCTRL_ACKEN_Getable(void)
功能描述	主控接收模式下，读取主机给从机回应使能位
输入参数 1	无
输出参数	无
返回值	主机给从机回应使能位： 0: 主机不响应从机 1: 主机发送回应 ACKDT 给从机

**21.2.12 I2C\_I2CCTRL\_RCEN\_Setable**

函数名	I2C_I2CCTRL_RCEN_Setable
函数原型	void I2C_I2CCTRL_RCEN_Setable(FunState NewState)
功能描述	主控接收模式下，设置接收使能位
输入参数 1	接收使能位设置： DISABLE：禁止接收 ENABLE：使能接收
输出参数	无
返回值	无

**21.2.13 I2C\_I2CCTRL\_RCEN\_Getable**

函数名	I2C_I2CCTRL_RCEN_Getable
函数原型	FunState I2C_I2CCTRL_RCEN_Getable(void)
功能描述	主控接收模式下，读取接收使能位
输入参数 1	无
输出参数	无
返回值	接收使能位设置： DISABLE：禁止接收 ENABLE：使能接收

**21.2.14 I2C\_I2CCTRL\_RSEN\_Setable**

函数名	I2C_I2CCTRL_RSEN_Setable
函数原型	void I2C_I2CCTRL_RSEN_Setable(FunState NewState)
功能描述	设置 Repeated START 时序产生使能控制位
输入参数 1	Repeated START 时序产生使能控制位： DISABLE：禁止 Repeated START 时序产生 ENABLE：使能 Repeated START 时序产生
输出参数	无
返回值	无

**21.2.15 I2C\_I2CCTRL\_PEN\_Setable**

函数名	I2C_I2CCTRL_PEN_Setable
函数原型	void I2C_I2CCTRL_PEN_Setable(FunState NewState)
功能描述	设置 STOP 时序产生使能控制位



输入参数 1	无
输出参数	无
返回值	STOPT 时序产生使能控制位： DISABLE：禁止 STOP 时序产生 ENABLE：使能 STOP 时序产生

### 21.2.16 I2C\_I2CCTRL\_SEN\_Setable

函数名	I2C_I2CCTRL_SEN_Setable
函数原型	void I2C_I2CCTRL_SEN_Setable(FunState NewState)
功能描述	设置 START 时序产生使能控制位
输入参数 1	START 时序产生使能控制位： DISABLE：禁止 START 时序产生 ENABLE：使能 START 时序产生
输出参数	无
返回值	无

### 21.2.17 I2C\_I2CBRG\_Write

函数名	I2C_I2CBRG_Write
函数原型	void I2C_I2CBRG_Write(uint32_t SetValue)
功能描述	设置波特率分频值
输入参数 1	波特率分频值：0-1F
输出参数	无
返回值	无

### 21.2.18 I2C\_I2CBRG\_Read

函数名	I2C_I2CBRG_Read
函数原型	uint32_t I2C_I2CBRG_Read(void)
功能描述	读取波特率分频值
输入参数 1	无
输出参数	无
返回值	波特率分频值：0-1F

### 21.2.19 I2C\_BaudREG\_Calc

函数名	I2C_BaudREG_Calc
-----	------------------

函数原型	uint32_t I2C_BaudREG_Calc(uint32_t I2CClk, uint32_t APBClk)
功能描述	I2C 波特率寄存器值计算
输入参数 1	期望波特率:0- APBClk
输入参数 2	APB 时钟频率
输出参数	无
返回值	无

### 21.2.20 I2C\_I2CSTA\_RW\_Chk

函数名	I2C_I2CSTA_RW_Chk
函数原型	FlagStatus I2C_I2CSTA_RW_Chk(void)
功能描述	读取 I2C 接口状态
输入参数 1	无
输出参数	无
返回值	I2C 接口状态: 0: 已完成数据传输 1: 接口处于读写状态, 正在进行数据传输

### 21.2.21 I2C\_I2CSTA\_BF\_Chk

函数名	I2C_I2CSTA_BF_Chk
函数原型	FlagStatus I2C_I2CSTA_BF_Chk(void)
功能描述	读取缓冲器满状态位
输入参数 1	无
输出参数	无
返回值	缓冲器满状态: 0: 缓冲器未满 1: 缓冲器满

### 21.2.22 I2C\_I2CSTA\_S\_Chk

函数名	I2C_I2CSTA_S_Chk
函数原型	FlagStatus I2C_I2CSTA_S_Chk(void)
功能描述	读取 START 标志位
输入参数 1	无
输出参数	无
返回值	START 标志: 0: 未检测到 START 时序

1: 检测到 START 时序

**21.2.23 I2C\_I2CSTA\_P\_Chk**

函数名	I2C_I2CSTA_P_Chk
函数原型	FlagStatus I2C_I2CSTA_P_Chk(void)
功能描述	读取 STOP 标志位
输入参数 1	无
输出参数	无
返回值	STOP 标志: 0: 未检测到 STOP 时序 1: 检测到 STOP 时序

**21.2.24 I2C\_I2CSTA\_WCOL\_Clr**

函数名	I2C_I2CSTA_WCOL_Clr
函数原型	void I2C_I2CSTA_WCOL_Clr(void)
功能描述	清除写冲突检测位
输入参数 1	无
输出参数	无
返回值	无

**21.2.25 I2C\_I2CSTA\_WCOL\_Chk**

函数名	I2C_I2CSTA_WCOL_Chk
函数原型	FlagStatus I2C_I2CSTA_WCOL_Chk(void)
功能描述	读取写冲突检测位
输入参数 1	无
输出参数	无
返回值	0: 未发生写冲突 1: 发生写冲突

**21.2.26 I2C\_I2CSTA\_ACKSTA\_Clr**

函数名	I2C_I2CSTA_ACKSTA_Clr
函数原型	void I2C_I2CSTA_ACKSTA_Clr(void)
功能描述	清除来自从机的回应信号
输入参数 1	无

输出参数	无
返回值	无

### 21.2.27 I2C\_I2CSTA\_ACKSTA\_Chk

函数名	I2C_I2CSTA_ACKSTA_Chk
函数原型	FlagStatus I2C_I2CSTA_ACKSTA_Chk(void)
功能描述	读取 来自从机的回应信号
输入参数 1	无
输出参数	无
返回值	来自从机的回应信号： 0：接收到从机的回应 0 1：从机无回应

### 21.2.28 I2C\_I2CSTA\_ACKDT\_Set

函数名	I2C_I2CSTA_ACKDT_Set
函数原型	void I2C_I2CSTA_ACKDT_Set(uint32_t SetValue)
功能描述	设置主机回应信号的状态
输入参数 1	主机回应信号的状态： 0：主机将给从机的回应为 0 1：主机将不给从机回应
输出参数	无
返回值	无

### 21.2.29 I2C\_I2CSTA\_ACKDT\_Get

函数名	I2C_I2CSTA_ACKDT_Get
函数原型	uint32_t I2C_I2CSTA_ACKDT_Get(void)
功能描述	读取主机回应信号的状态设置
输入参数 1	无
输出参数	无
返回值	主机回应信号的状态： 0：主机将给从机的回应为 0 1：主机将不给从机回应



**21.2.30 I2C\_I2CIR\_I2CIE\_Setable**

函数名	I2C_I2CIR_I2CIE_Setable
函数原型	void I2C_I2CIR_I2CIE_Setable(FunState NewState)
功能描述	设置 I2C 中断使能控制位
输入参数 1	I2C 中断使能控制位： DISABLE: 禁止 I2C 中断 ENABLE: 使能 I2C 中断
输出参数	无
返回值	无

**21.2.31 I2C\_I2CIR\_I2CIE\_Getable**

函数名	I2C_I2CIR_I2CIE_Getable
函数原型	FunState I2C_I2CIR_I2CIE_Getable(void)
功能描述	读取 I2C 中断使能控制位设置
输入参数 1	无
输出参数	无
返回值	I2C 中断使能控制位： DISABLE: 禁止 I2C 中断 ENABLE: 使能 I2C 中断

**21.2.32 I2C\_I2CIR\_I2CIF\_Clr**

函数名	I2C_I2CIR_I2CIF_Clr
函数原型	void I2C_I2CIR_I2CIF_Clr(void)
功能描述	清除 I2C 中断标志位
输入参数 1	无
输出参数	无
返回值	无

**21.2.33 I2C\_I2CIR\_I2CIF\_Chk**

函数名	I2C_I2CIR_I2CIF_Chk
函数原型	FlagStatus I2C_I2CIR_I2CIF_Chk(void)
功能描述	读取 I2C 中断标志位
输入参数 1	无
输出参数	无

返回值	I2C 中断标志: 0: 未产生 I2C 中断 1: 产生 I2C 中断
-----	--------------------------------------------

### 21.2.34 I2C\_I2CFSM\_I2CFSM\_Get

函数名	I2C_I2CFSM_I2CFSM_Get
函数原型	uint32_t I2C_I2CFSM_I2CFSM_Get(void)
功能描述	读取 I2C 主状态机编码
输入参数 1	无
输出参数	无
返回值	I2C 主状态机编码: 0000: IDLE 0001: START1 0010: START2 0011: START_DONE 0100: TX_STATE 0101: RXACK 0110: OP_IDLE 0111: STOP1 1000: STOP2 1001: STOP_DONE 1010: RTP_START 1011: RX_STATE 1100: ACK_STATE 1101: STOP0

### 21.2.35 I2C\_I2CERR\_ERRIE\_Setable

函数名	I2C_I2CERR_ERRIE_Setable
函数原型	void I2C_I2CERR_ERRIE_Setable(FunState NewState)
功能描述	设置错误标志中断使能
输入参数 1	错误标志中断使能: DISABLE: 禁止 I2C 中断 ENABLE: 使能 I2C 中断
输出参数	无
返回值	无

**21.2.36 I2C\_I2CERR\_ERRIE\_Getable**

函数名	I2C_I2CERR_ERRIE_Getable
函数原型	FunState I2C_I2CERR_ERRIE_Getable(void)
功能描述	读取错误标志中断使能设置
输入参数 1	无
输出参数	无
返回值	错误标志中断使能： DISABLE：禁止 I2C 中断 ENABLE：使能 I2C 中断

**21.2.37 I2C\_I2CERR\_OIERR\_Clr**

函数名	I2C_I2CERR_OIERR_Clr
函数原型	void I2C_I2CERR_OIERR_Clr(void)
功能描述	清除 OP_IDLE 状态下错误标志位
输入参数 1	无
输出参数	无
返回值	无

**21.2.38 I2C\_I2CERR\_OIERR\_Chk**

函数名	I2C_I2CERR_OIERR_Chk
函数原型	FlagStatus I2C_I2CERR_OIERR_Chk(void)
功能描述	读取 OP_IDLE 状态下错误标志位
输入参数 1	无
输出参数	无
返回值	OP_IDLE 状态下错误标志： 0： OP_IDLE 状态下未发生错误 1： OP_IDLE 状态下发生错误

**21.2.39 I2C\_I2CERR\_SDERR\_Clr**

函数名	I2C_I2CERR_SDERR_Clr
函数原型	void I2C_I2CERR_SDERR_Clr(void)
功能描述	清除 START_DONE 状态下错误标志位
输入参数 1	无
输出参数	无

返回值	无
-----	---

#### 21.2.40 I2C\_I2CERR\_SDERR\_Chk

函数名	I2C_I2CERR_SDERR_Chk
函数原型	FlagStatus I2C_I2CERR_SDERR_Chk(void)
功能描述	读取 START_DONE 状态下错误标志位
输入参数 1	无
输出参数	无
返回值	START_DONE 状态下错误标志： 0: START_DONE 状态下未产生错误 1: START_DONE 状态下产生错误

#### 21.2.41 I2C\_I2CERR\_IERR\_Clr

函数名	I2C_I2CERR_IERR_Clr
函数原型	void I2C_I2CERR_IERR_Clr(void)
功能描述	清除 IDLE 状态下错误标志位
输入参数 1	无
输出参数	无
返回值	无

#### 21.2.42 I2C\_I2CERR\_IERR\_Chk

函数名	I2C_I2CERR_IERR_Chk
函数原型	FlagStatus I2C_I2CERR_IERR_Chk(void)
功能描述	读取 IDLE 状态下错误标志位
输入参数 1	无
输出参数	无
返回值	IDLE 状态下错误标志： 0: IDLE 状态下未产生错误 1: IDLE 状态下产生错误

## 22 SPI

### 22.1 SPI 寄存器结构

寄存器	描述
SPI1CR1	SPI 控制寄存器 1
SPI2CR1	
SPI1CR2	SPI 控制寄存器 2
SPI2CR2	
SPI1CR3	SPI 控制寄存器 3
SPI2CR3	
SPI1IE	SPI 中断控制寄存器
SPI2IE	
SPI1IF	SPI 中断标志位寄存器
SPI2IF	
SPI1TXBUF	SPI 发送缓冲寄存器
SPI2 TXBUF	
SPI1RXBUF	SPI 接收缓冲寄存器
SPI2 RXBUF	

### 22.2 SPI 库函数

序号	函数名	描述
1	SPIx_Deinit	SPI 初始化为缺省值
2	SPI_Master_Init	SPI 主模式初始化
3	SPI_Slave_Init	SPI 从模式初始化
4	SPIx_SPICR2_SPIEN_Setable	设置 SPI 使能
5	SPIx_SPICR2_SPIEN_Getable	读取 SPI 使能的设置
6	SPIx_SPICR3_TXBFC_Clr	清除发送 BUFFER
7	SPIx_SPICR3_RXBFC_Clr	清除接收 BUFFER
8	SPIx_SPICR3_MERRC_Clr	清除主模式错误
9	SPIx_SPICR3_SERRC_Clr	清除从模式错误
10	SPIx_SPIIE_ERRIE_Setable	设置 HSPI 错误中断使能
11	SPIx_SPIIE_ERRIE_Getable	读取 HSPI 错误中断使能的设置
12	SPIx_SPIIE_TXIE_Setable	设置发送中断使能



13	SPIx_SPIIE_TXIE_Getable	读取发送中断使能的设置
14	SPIx_SPIIE_RXIE_Setable	设置接收中断使能
15	SPIx_SPIIE_RXIE_Getable	读取接收中断使能
16	SPI_SSN_Set_Low	设置 SPI 的 SSN 为低电平
17	SPI_SSN_Set_High	设置 SPI 的 SSN 为高电平
18	SPI_Send_Byte	SPI 接收一个字节
19	SPI_Recv_Byte	SPI 发送一个字节
20	SPI_RW_Byte	SPI 发送和接收一个字节
21	SPIx_SPITXBUF_Write	写 SPI 发送缓存
22	SPIx_SPITXBUF_Read	读取 SPI 发送缓存
23	SPIx_SPIRXBUF_Write	写 SPI 接收缓存
24	SPIx_SPIRXBUF_Read	读取 SPI 接收缓存
25	SPIx_SPICR1_CPHOL_Set	设置时钟极性
26	SPIx_SPICR1_CPHOL_Get	读取时钟极性设置
27	SPIx_SPICR1_CPHA_Set	设置时钟相位
28	SPIx_SPICR1_CPHA_Get	读取时钟相位设置
29	SPIx_SPICR1_LSBF_Set	设置帧格式
30	SPIx_SPICR1_LSBF_Get	读取帧格式的设置
31	SPIx_SPICR1_MM_Set	设置 MASTER/SLAVE 模式选择
32	SPIx_SPICR1_MM_Get	读取 MASTER/SLAVE 模式选择
33	SPIx_SPICR1_WAIT_Set	Master 模式下设置加入 SCK cycle 的数值
34	SPIx_SPICR1_WAIT_Get	读取 Master 模式下设置加入 SCK cycle 的数值
35	SPIx_SPICR1_BAUD_Set	MASTER 模式设置波特率
36	SPIx_SPICR1_BAUD_Get	读取 MASTER 模式波特率的设置
37	SPIx_SPICR2_SSNM_Set	Master 模式下 SSN 控制模式的选择
38	SPIx_SPICR2_SSNM_Get	读取 Master 模式下 SSN 控制模式选择
39	SPIx_SPICR2_SSN_Set	Master 模式下设置软件控制 SSN 输出电平
40	SPIx_SPICR2_SSN_Get	读取 Master 模式下软件控制 SSN 输出电平
41	SPIx_SPICR2_SSNSSEN_Set	Master 模式下设置 SSN 的控制方式
42	SPIx_SPICR2_SSNSSEN_Get	读取 Master 模式下 SSN 的控制方



		式
43	SPIx_SPICR1_MSPA_Set	设置 Master 对 MISO 信号的采样位置调整
44	SPIx_SPICR1_MSPA_Get	读取 Master 对 MISO 信号的采样位置调整
45	SPIx_SPICR1_SSPA_Set	设置 Slave MISO 发送位置调整
46	SPIx_SPICR1_SSPA_Get	读取 Slave MISO 发送位置调整
47	SPIx_SPICR2_FLTEN_Setable	设置 Slave 输入管脚滤波使能
48	SPIx_SPICR2_FLTEN_Getable	读取 Slave 输入管脚滤波使能状态
49	SPIx_SPIIF_TXBE_Chk	读取 TX Buffer Empty 标志位
50	SPIx_SPIIF_RXBF_Chk	读取 RX Buffer Full 标志位
51	SPIx_SPIIF_BUSY_Chk	读取 SPI 空闲标志
52	SPIx_SPIIF_TXCOL_Clr	清除发送缓存溢出标志
53	SPIx_SPIIF_TXCOL_Chk	读取发送缓存溢出标志
54	SPIx_SPIIF_RXCOL_Clr	清除接收缓存溢出标志
55	SPIx_SPIIF_RXCOL_Chk	读取接收缓存溢出标志
56	SPIx_SPIIF_MERR_Chk	读取 Master Error 标志
57	SPIx_SPIIF_SERR_Chk	读取 Slave Error 标志
58	SPIx_SPICR2_TXO_AC_Set	设置 TXONLY 硬件自动清空的使能
59	SPIx_SPICR2_TXO_AC_Get	读取 TXONLY 硬件自动清空的使能状态
60	SPIx_SPICR2_TXO_Setable	设置 TTXONLY 控制位
61	SPIx_SPICR2_TXO_Getable	读取 TTXONLY 控制位设置

### 22.2.1 SPIx\_Deinit

函数名	SPIx_Deinit
函数原型	void SPIx_Deinit(SPIx_Type* SPIx)
功能描述	SPI 初始化为缺省值
输入参数 1	SPI1/SPI2
输出参数	无
返回值	无

### 22.2.2 SPI\_Master\_Init

函数名	SPI_Master_Init
函数原型	void SPI_Master_Init(SPIx_Type* SPIx, SPI_Master_SInitTypeDef* para)

功能描述	SPI 主模式初始化						
输入参数 1	SPI1/SPI2						
输入参数 2	<div>SPI_Master 参数:</div> <table><tr><td>波特率</td></tr><tr><td>高低位发送顺序: 0: 先发低位 1: 先发高位</td></tr><tr><td>时钟极性: 0: 串行时钟停止在低电平 1: 串行时钟停止在高电平</td></tr><tr><td>时钟相位选择: 0: 第一个时钟边沿时第一个捕捉边沿 1: 第二个时钟边沿时第一个捕捉边沿</td></tr><tr><td>Master 模式下 SSN 控制模式选择: 0: 每次发完 8bit 后 Master 保持 SSN 为低 1: 每次发完 8bit 后 Master 拉高 SSN</td></tr><tr><td>Master 模式下,SSN 的控制源: 0: 硬件自动控制 1: 软件控制</td></tr></table>	波特率	高低位发送顺序: 0: 先发低位 1: 先发高位	时钟极性: 0: 串行时钟停止在低电平 1: 串行时钟停止在高电平	时钟相位选择: 0: 第一个时钟边沿时第一个捕捉边沿 1: 第二个时钟边沿时第一个捕捉边沿	Master 模式下 SSN 控制模式选择: 0: 每次发完 8bit 后 Master 保持 SSN 为低 1: 每次发完 8bit 后 Master 拉高 SSN	Master 模式下,SSN 的控制源: 0: 硬件自动控制 1: 软件控制
波特率							
高低位发送顺序: 0: 先发低位 1: 先发高位							
时钟极性: 0: 串行时钟停止在低电平 1: 串行时钟停止在高电平							
时钟相位选择: 0: 第一个时钟边沿时第一个捕捉边沿 1: 第二个时钟边沿时第一个捕捉边沿							
Master 模式下 SSN 控制模式选择: 0: 每次发完 8bit 后 Master 保持 SSN 为低 1: 每次发完 8bit 后 Master 拉高 SSN							
Master 模式下,SSN 的控制源: 0: 硬件自动控制 1: 软件控制							
输出参数	无						
返回值	无						

### 22.2.3 SPI\_Slave\_Init

函数名	SPI_Slave_Init
函数原型	void SPI_Slave_Init(SPIx_Type* SPIx, SPI_Slave_SInitTypeDef* para)
功能描述	SPI 从模式初始化
输入参数 1	SPI1/SPI2
输入参数 2	<div>SPI_Slave 参数:</div> <div><div>高低位发送顺序: 0: 先发低位 1: 先发高位</div><div>时钟极性: 0: 串行时钟停止在低电平 1: 串行时钟停止在高电平</div><div>时钟相位选择: 0: 第一个时钟边沿时第一个捕捉边沿 1: 第二个时钟边沿时第一个捕捉边沿</div></div>
输出参数	无
返回值	无



**22.2.4 SPIx\_SPICR2\_SPIEN\_Setable**

函数名	SPIx_SPICR2_SPIEN_Setable
函数原型	void SPIx_SPICR2_SPIEN_Setable(SPIx_Type* SPIx, FunState NewState)
功能描述	设置 SPI 使能
输入参数 1	SPI1/SPI2
输入参数 2	DISABLE: 禁止 SPI, 清空发送接收缓存 ENABLE: 使能 SPI
输出参数	无
返回值	无

**22.2.5 SPIx\_SPICR2\_SPIEN\_Getable**

函数名	SPIx_SPICR2_SPIEN_Getable
函数原型	FunState SPIx_SPICR2_SPIEN_Getable(SPIx_Type* SPIx)
功能描述	读取 SPI 使能的设置
输入参数 1	SPI1/SPI2
输出参数	无
返回值	DISABLE: 禁止 SPI, 清空发送接收缓存 ENABLE: 使能 SPI

**22.2.6 SPIx\_SPICR3\_TXBFC\_Clr**

函数名	SPIx_SPICR3_TXBFC_Clr
函数原型	void SPIx_SPICR3_TXBFC_Clr(SPIx_Type* SPIx)
功能描述	清除发送 BUFFER
输入参数 1	SPI1/SPI2
输出参数	无
返回值	无

**22.2.7 SPIx\_SPICR3\_RXBFC\_Clr**

函数名	SPIx_SPICR3_RXBFC_Clr
函数原型	void SPIx_SPICR3_RXBFC_Clr(SPIx_Type* SPIx)
功能描述	清除接收 BUFFER
输入参数 1	SPI1/SPI2
输出参数	无
返回值	无

### 22.2.8 SPIx\_SPICR3\_MERRC\_Clr

函数名	SPIx_SPICR3_MERRC_Clr
函数原型	void SPIx_SPICR3_MERRC_Clr(SPIx_Type* SPIx)
功能描述	清除主模式错误
输入参数 1	SPI1/SPI2
输出参数	无
返回值	无

### 22.2.9 SPIx\_SPICR3\_SERRC\_Clr

函数名	SPIx_SPICR3_SERRC_Clr
函数原型	void SPIx_SPICR3_SERRC_Clr(SPIx_Type* SPIx)
功能描述	清除从模式错误
输入参数 1	SPI1/SPI2
输出参数	无
返回值	无

### 22.2.10 SPIx\_SPIIE\_ERRIE\_Setable

函数名	SPIx_SPIIE_ERRIE_Setable
函数原型	void SPIx_SPIIE_ERRIE_Setable(SPIx_Type* SPIx, FunState NewState)
功能描述	设置 HSPI 错误中断使能
输入参数 1	SPI1/SPI2
输入参数 2	HSPI 错误中断使能： DISABLE：禁止 HSPI 错误中断 ENABLE：使能 HSPI 错误中断
输出参数	无
返回值	无

### 22.2.11 SPIx\_SPIIE\_ERRIE\_Getable

函数名	SPIx_SPIIE_ERRIE_Getable
函数原型	FunState SPIx_SPIIE_ERRIE_Getable(SPIx_Type* SPIx)
功能描述	读取 HSPI 错误中断使能的设置
输入参数 1	SPI1/SPI2
输出参数	无

返回值	HSPI 错误中断使能： DISABLE：禁止 HSPI 错误中断 ENABLE：使能 HSPI 错误中断
-----	-------------------------------------------------------------

### 22.2.12 SPIx\_SPIIE\_TXIE\_Setable

函数名	SPIx_SPIIE_TXIE_Setable
函数原型	void SPIx_SPIIE_TXIE_Setable(SPIx_Type* SPIx, FunState NewState)
功能描述	设置发送中断使能
输入参数 1	SPI1/SPI2
输入参数 2	发送中断使能： DISABLE：禁止发送中断 ENABLE：使能发送中断
输出参数	无
返回值	无

### 22.2.13 SPIx\_SPIIE\_TXIE\_Getable

函数名	SPIx_SPIIE_TXIE_Getable
函数原型	FunState SPIx_SPIIE_TXIE_Getable(SPIx_Type* SPIx)
功能描述	读取发送中断使能的设置
输入参数 1	SPI1/SPI2
输出参数	无
返回值	发送中断使能： DISABLE：禁止发送中断 ENABLE：使能发送中断

### 22.2.14 SPIx\_SPIIE\_RXIE\_Setable

函数名	SPIx_SPIIE_RXIE_Setable
函数原型	void SPIx_SPIIE_RXIE_Setable(SPIx_Type* SPIx, FunState NewState)
功能描述	设置接收中断使能
输入参数 1	SPI1/SPI2
输入参数 2	接收中断使能： DISABLE：禁止接收中断 ENABLE：使能接收中断
输出参数	无
返回值	无

### 22.2.15 SPIx\_SPIIE\_RXIE\_Getable

函数名	SPIx_SPIIE_RXIE_Getable
函数原型	FunState SPIx_SPIIE_RXIE_Getable(SPIx_Type* SPIx)
功能描述	读取接收中断使能
输入参数 1	SPI1/SPI2
输出参数	无
返回值	接收中断使能： DISABLE：禁止发送中断 ENABLE：使能发送中断

### 22.2.16 SPI\_SSN\_Set\_Low

函数名	SPI_SSN_Set_Low
函数原型	void SPI_SSN_Set_Low(SPIx_Type* SPIx)
功能描述	设置 SPI 的 SSN 为低电平
输入参数 1	SPI1/SPI2
输出参数	无
返回值	无

### 22.2.17 SPI\_SSN\_Set\_High

函数名	SPI_SSN_Set_High
函数原型	void SPI_SSN_Set_High(SPIx_Type* SPIx)
功能描述	设置 SPI 的 SSN 为高电平
输入参数 1	SPI1/SPI2
输出参数	无
返回值	无

### 22.2.18 SPI\_Send\_Byte

函数名	SPI_Send_Byte
函数原型	void SPI_Send_Byte(SPIx_Type* SPIx, unsigned char data)
功能描述	SPI 发送一个字节
输入参数 1	SPI1/SPI2
输入参数 2	待发送的数据:0-FF
输出参数	无



返回值	无
-----	---

### 22.2.19 SPI\_Rcv\_Byte

函数名	SPI_Rcv_Byte
函数原型	unsigned char SPI_Rcv_Byte(SPIx_Type* SPIx)
功能描述	SPI 接收一个字节
输入参数 1	SPI1/SPI2
输出参数	无
返回值	接收的数据:0-FF

### 22.2.20 SPI\_RW\_Byte

函数名	SPI_RW_Byte
函数原型	unsigned char SPI_RW_Byte(SPIx_Type* SPIx, unsigned char data)
功能描述	SPI 发送和接收一个字节
输入参数 1	SPI1/SPI2
输入参数 2	待发送的数据:0-FF
输出参数	无
返回值	接收的数据:0-FF

### 22.2.21 SPIx\_SPITXBUF\_Write

函数名	SPIx_SPITXBUF_Write
函数原型	void SPIx_SPITXBUF_Write(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	写 SPI 发送缓存
输入参数 1	SPI1/SPI2
输入参数 2	SPI 发送缓存数据:0-FF
输出参数	无
返回值	无

### 22.2.22 SPIx\_SPITXBUF\_Read

函数名	SPIx_SPITXBUF_Read
函数原型	uint32_t SPIx_SPITXBUF_Read(SPIx_Type* SPIx)
功能描述	读取 SPI 发送缓存
输入参数 1	SPI1/SPI2
输出参数	无

返回值	SPI 发送缓存的值:0-FF
-----	-----------------

### 22.2.23 SPIx\_SPIRXBUF\_Write

函数名	SPIx_SPIRXBUF_Write
函数原型	void SPIx_SPIRXBUF_Write(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	写 SPI 接收缓存
输入参数 1	SPI1/SPI2
输入参数 2	SPI 接收缓存数据:0-FF
输出参数	无
返回值	无

### 22.2.24 SPIx\_SPIRXBUF\_Read

函数名	SPIx_SPIRXBUF_Read
函数原型	uint32_t SPIx_SPIRXBUF_Read(SPIx_Type* SPIx)
功能描述	读取 SPI 接收缓存
输入参数 1	SPI1/SPI2
输出参数	无
返回值	SPI 接收缓存的值:0-FF

### 22.2.25 SPIx\_SPICR1\_CPHOL\_Set

函数名	SPIx_SPICR1_CPHOL_Set
函数原型	void SPIx_SPICR1_CPHOL_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	设置时钟极性
输入参数 1	SPI1/SPI2
输入参数 2	时钟极性: 0: 串行时钟停止在低电平 1: 串行时钟停止在高电平
输出参数	无
返回值	无

### 22.2.26 SPIx\_SPICR1\_CPHOL\_Get

函数名	SPIx_SPICR1_CPHOL_Get
函数原型	uint32_t SPIx_SPICR1_CPHOL_Get(SPIx_Type* SPIx)
功能描述	读取时钟极性设置

输入参数 1	SPI1/SPI2
输出参数	无
返回值	时钟极性： 0：串行时钟停止在低电平 1：串行时钟停止在高电平

### 22.2.27 SPIx\_SPICR1\_CPHA\_Set

函数名	SPIx_SPICR1_CPHA_Set
函数原型	void SPIx_SPICR1_CPHA_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	设置时钟相位
输入参数 1	SPI1/SPI2
输入参数 2	时钟相位选择： 0：第一个时钟边沿时第一个捕捉边沿 1：第二个时钟边沿时第一个捕捉边沿
输出参数	无
返回值	无

### 22.2.28 SPIx\_SPICR1\_CPHA\_Get

函数名	SPIx_SPICR1_CPHA_Get
函数原型	uint32_t SPIx_SPICR1_CPHA_Get(SPIx_Type* SPIx)
功能描述	读取时钟相位设置
输入参数 1	SPI1/SPI2
输出参数	无
返回值	时钟相位选择： 0：第一个时钟边沿时第一个捕捉边沿 1：第二个时钟边沿时第一个捕捉边沿

### 22.2.29 SPIx\_SPICR1\_LSBF\_Set

函数名	SPIx_SPICR1_LSBF_Set
函数原型	void SPIx_SPICR1_LSBF_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	设置帧格式
输入参数 1	SPI1/SPI2
输入参数 2	帧格式： 0：先发 LSB 1：先发 MSB

输出参数	无
返回值	无

### 22.2.30 SPIx\_SPICR1\_LSBF\_Get

函数名	SPIx_SPICR1_LSBF_Get
函数原型	uint32_t SPIx_SPICR1_LSBF_Get(SPIx_Type* SPIx)
功能描述	读取帧格式的设置
输入参数 1	SPI1/SPI2
输入参数 2	无
返回值	帧格式： 0：先发 LSB 1：先发 MSB

### 22.2.31 SPIx\_SPICR1\_MM\_Set

函数名	SPIx_SPICR1_MM_Set
函数原型	void SPIx_SPICR1_MM_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	设置 MASTER/SLAVE 模式选择
输入参数 1	SPI1/SPI2
输入参数 2	MASTER/SLAVE 模式选择： 0：Slave 模式 1：Master 模式
输出参数	无
返回值	无

### 22.2.32 SPIx\_SPICR1\_MM\_Get

函数名	SPIx_SPICR1_MM_Get
函数原型	uint32_t SPIx_SPICR1_MM_Get(SPIx_Type* SPIx)
功能描述	读取 MASTER/SLAVE 模式选择
输入参数 1	SPI1/SPI2
输出参数	无
返回值	MASTER/SLAVE 模式选择： 0：Slave 模式 1：Master 模式



**22.2.33 SPIx\_SPICR1\_WAIT\_Set**

函数名	SPIx_SPICR1_WAIT_Set
函数原型	void SPIx_SPICR1_WAIT_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	Master 模式下设置加入 SCK cycle 的数值
输入参数 1	SPI1/SPI2
输入参数 2	SCK cycle 的 WAIT 数值: SPIx_SPICR1_WAIT_WAIT_1: 每发完 8bit 至少加入 1 个 sck 周期等待时间再传输下一个 8Bit 数据 SPIx_SPICR1_WAIT_WAIT_2 每发完 8bit 至少加入 2 个 sck 周期等待时间再传输下一个 8Bit 数据 SPIx_SPICR1_WAIT_WAIT_3: 每发完 8bit 至少加入 3 个 sck 周期等待时间再传输下一个 8Bit 数据 SPIx_SPICR1_WAIT_WAIT_4: 每发完 8bit 至少加入 4 个 sck 周期等待时间再传输下一个 8Bit 数据
输出参数	无
返回值	无

**22.2.34 SPIx\_SPICR1\_WAIT\_Get**

函数名	SPIx_SPICR1_WAIT_Get
函数原型	uint32_t SPIx_SPICR1_WAIT_Get(SPIx_Type* SPIx)
功能描述	读取 Master 模式下设置加入 SCK cycle 的数值
输入参数 1	SPI1/SPI2
输出参数	无
返回值	SCK cycle 的 WAIT 数值: SPIx_SPICR1_WAIT_WAIT_1: 每发完 8bit 至少加入 1 个 sck 周期等待时间再传输下一个 8Bit 数据 SPIx_SPICR1_WAIT_WAIT_2 每发完 8bit 至少加入 2 个 sck 周期等待时间再传输下一个 8Bit 数据 SPIx_SPICR1_WAIT_WAIT_3: 每发完 8bit 至少加入 3 个 sck 周期等待时间再传输下一个 8Bit 数据 SPIx_SPICR1_WAIT_WAIT_4: 每发完 8bit 至少加入 4 个 sck 周期等待时间再传输下一个 8Bit 数据

**22.2.35 SPIx\_SPICR1\_BAUD\_Set**

函数名	SPIx_SPICR1_BAUD_Set
-----	----------------------

函数原型	void SPIx_SPICR1_BAUD_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	MASTER 模式配置波特率分频值
输入参数 1	SPI1/SPI2
输入参数 2	MASTER 模式波特率分频值： 000: fAHBCLK/2 001: fAHBCLK/4 010: fAHBCLK/8 011: fAHBCLK/16 100: fAHBCLK/32 101: fAHBCLK/64 110: fAHBCLK/128 111: fAHBCLK/256
输出参数	无
返回值	无

### 22.2.36 SPIx\_SPICR1\_BAUD\_Get

函数名	SPIx_SPICR1_BAUD_Get
函数原型	uint32_t SPIx_SPICR1_BAUD_Get(SPIx_Type* SPIx)
功能描述	读取 MASTER 模式波特率分频值
输入参数 1	SPI1/SPI2
输出参数	无
返回值	MASTER 模式波特率分频值： 000: fAHBCLK/2 001: fAHBCLK/4 010: fAHBCLK/8 011: fAHBCLK/16 100: fAHBCLK/32 101: fAHBCLK/64 110: fAHBCLK/128 111: fAHBCLK/256

### 22.2.37 SPIx\_SPICR2\_SSNM\_Set

函数名	SPIx_SPICR2_SSNM_Set
函数原型	void SPIx_SPICR2_SSNM_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	Master 模式下 SSN 控制模式的选择
输入参数 1	SPI1/SPI2

输入参数 2	Master 模式下 SSN 控制模式选择： 0：每次发完 8bit 后 Master 保持 SSN 为低 1：每次发完 8bit 后 Master 拉高 SSN
输出参数	无
返回值	无

### 22.2.38 SPIx\_SPICR2\_SSNM\_Get

函数名	SPIx_SPICR2_SSNM_Get
函数原型	uint32_t SPIx_SPICR2_SSNM_Get(SPIx_Type* SPIx)
功能描述	读取 Master 模式下 SSN 控制模式选择
输入参数 1	SPI1/SPI2
输出参数	无
返回值	Master 模式下 SSN 控制模式选择： 0：每次发完 8bit 后 Master 保持 SSN 为低 1：每次发完 8bit 后 Master 拉高 SSN

### 22.2.39 SPIx\_SPICR2\_SSN\_Set

函数名	SPIx_SPICR2_SSN_Set
函数原型	void SPIx_SPICR2_SSN_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	Master 模式下设置软件控制 SSN 输出电平
输入参数 1	SPI1/SPI2
输入参数 2	Master 模式下软件控制 SSN 输出电平： 0：SSN 输出低电平 1：SSN 输出高电平
输出参数	无
返回值	无

### 22.2.40 SPIx\_SPICR2\_SSN\_Get

函数名	SPIx_SPICR2_SSN_Get
函数原型	uint32_t SPIx_SPICR2_SSN_Get(SPIx_Type* SPIx)
功能描述	读取 Master 模式下软件控制 SSN 输出电平
输入参数 1	SPI1/SPI2
输出参数	无
返回值	Master 模式下软件控制 SSN 输出电平： 0：SSN 输出低电平

1: SSN 输出高电平

**22.2.41 SPIx\_SPICR2\_SSNSSEN\_Set**

函数名	SPIx_SPICR2_SSNSSEN_Set
函数原型	void SPIx_SPICR2_SSNSSEN_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	Master 模式下设置 SSN 的控制方式
输入参数 1	SPI1/SPI2
输入参数 2	Master 模式下,SSN 的控制模式: 0: 硬件自动控制 1: 软件控制
输出参数	无
返回值	无

**22.2.42 SPIx\_SPICR2\_SSNSSEN\_Get**

函数名	SPIx_SPICR2_SSNSSEN_Get
函数原型	uint32_t SPIx_SPICR2_SSNSSEN_Get(SPIx_Type* SPIx)
功能描述	读取 Master 模式下 SSN 的控制方式
输入参数 1	SPI1/SPI2
输出参数	无
返回值	Master 模式下,SSN 的控制模式: 0: 硬件自动控制 1: 软件控制

**22.2.43 SPIx\_SPICR1\_MSPA\_Set**

函数名	SPIx_SPICR1_MSPA_Set
函数原型	void SPIx_SPICR1_MSPA_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	设置 Master 对 MISO 信号的采样位置调整
输入参数 1	SPI1/SPI2
输入参数 2	采样位置调整的逻辑值: 0: 不调整 1: 采样点延迟半个 SCK 周期
输出参数	无
返回值	无

**22.2.44 SPIx\_SPICR1\_MSPA\_Get**

函数名	SPIx_SPICR1_MSPA_Get
函数原型	uint32_t SPIx_SPICR1_MSPA_Get(SPIx_Type* SPIx)
功能描述	读取 Master 对 MISO 信号的采样位置调整
输入参数 1	SPI1/SPI2
输出参数	无
返回值	采样位置调整的逻辑值： 0：不调整 1：采样点延迟半个 SCK 周期

**22.2.45 SPIx\_SPICR1\_SSPA\_Set**

函数名	SPIx_SPICR1_SSPA_Set
函数原型	void SPIx_SPICR1_SSPA_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	设置 Slave MISO 发送位置调整
输入参数 1	SPI1/SPI2
输入参数 2	Slave MISO 发送位置调整的逻辑值： 0：不调整 1：提前半个 SCK 周期发送
输出参数	无
返回值	无

**22.2.46 SPIx\_SPICR1\_SSPA\_Get**

函数名	SPIx_SPICR1_SSPA_Get
函数原型	uint32_t SPIx_SPICR1_SSPA_Get(SPIx_Type* SPIx)
功能描述	读取 Slave MISO 发送位置调整
输入参数 1	SPI1/SPI2
输出参数	无
返回值	Slave MISO 发送位置调整的逻辑值： 0：不调整 1：提前半个 SCK 周期发送

**22.2.47 SPIx\_SPICR2\_FLTEN\_Setable**

函数名	SPIx_SPICR2_FLTEN_Setable
函数原型	void SPIx_SPICR2_FLTEN_Setable(SPIx_Type* SPIx, FunState NewState)



功能描述	设置 Slave 输入管脚滤波使能
输入参数 1	SPI1/SPI2
输入参数 2	Slave 输入管脚滤波使能： 0：不滤波 ENABLE：使能 4ns 滤波
输出参数	无
返回值	无

#### 22.2.48 SPIx\_SPICR2\_FLTEN\_Getable

函数名	SPIx_SPICR2_FLTEN_Getable
函数原型	FunState SPIx_SPICR2_FLTEN_Getable(SPIx_Type* SPIx)
功能描述	读取 Slave 输入管脚滤波使能状态
输入参数 1	SPI1/SPI2
输出参数	无
返回值	Slave 输入管脚滤波使能： 0：不滤波 ENABLE：使能 4ns 滤波

#### 22.2.49 SPIx\_SPIIF\_TXBE\_Chk

函数名	SPIx_SPIIF_TXBE_Chk
函数原型	FlagStatus SPIx_SPIIF_TXBE_Chk(SPIx_Type* SPIx)
功能描述	读取 TX Buffer Empty 标志位
输入参数 1	SPI1/SPI2
输出参数	无
返回值	TX Buffer Empty 标志位： 0：发送缓存满 1：发送缓存空

#### 22.2.50 SPIx\_SPIIF\_RXBF\_Chk

函数名	SPIx_SPIIF_RXBF_Chk
函数原型	FlagStatus SPIx_SPIIF_RXBF_Chk(SPIx_Type* SPIx)
功能描述	读取 RX Buffer Full 标志位
输入参数 1	SPI1/SPI2
输出参数	无
返回值	RX Buffer Full 标志位：

0: 接收缓存空  
1: 接收缓存满

### 22.2.51 SPIx\_SPIIF\_BUSY\_Chk

函数名	SPIx_SPIIF_BUSY_Chk
函数原型	FlagStatus SPIx_SPIIF_BUSY_Chk(SPIx_Type* SPIx)
功能描述	读取 SPI 空闲标志
输入参数 1	SPI1/SPI2
输出参数	无
返回值	SPI 空闲标志: 0: SPI 忙 1: SPI 空闲

### 22.2.52 SPIx\_SPIIF\_TXCOL\_Clr

函数名	SPIx_SPIIF_TXCOL_Clr
函数原型	void SPIx_SPIIF_TXCOL_Clr(SPIx_Type* SPIx)
功能描述	清除发送缓存溢出标志
输入参数 1	SPI1/SPI2
输出参数	无
返回值	无

### 22.2.53 SPIx\_SPIIF\_TXCOL\_Chk

函数名	SPIx_SPIIF_TXCOL_Chk
函数原型	FlagStatus SPIx_SPIIF_TXCOL_Chk(SPIx_Type* SPIx)
功能描述	读取发送缓存溢出标志
输入参数 1	SPI1/SPI2
输出参数	无
返回值	发送缓存溢出标志: 0: 发送缓存未溢出 1: 发送缓存溢出

### 22.2.54 SPIx\_SPIIF\_RXCOL\_Clr

函数名	SPIx_SPIIF_RXCOL_Clr
函数原型	void SPIx_SPIIF_RXCOL_Clr(SPIx_Type* SPIx)



功能描述	清除接收缓存溢出标志
输入参数 1	SPI1/SPI2
输出参数	无
返回值	无

### 22.2.55 SPIx\_SPIIF\_RXCOL\_Chk

函数名	SPIx_SPIIF_RXCOL_Chk
函数原型	FlagStatus SPIx_SPIIF_RXCOL_Chk(SPIx_Type* SPIx)
功能描述	读取接收缓存溢出标志
输入参数 1	SPI1/SPI2
输出参数	无
返回值	接收缓存溢出标志： 0：接收缓存未溢出 1：接收缓存溢出

### 22.2.56 SPIx\_SPIIF\_MERR\_Chk

函数名	SPIx_SPIIF_MERR_Chk
函数原型	FlagStatus SPIx_SPIIF_MERR_Chk(SPIx_Type* SPIx)
功能描述	读取 Master Error 标志
输入参数 1	SPI1/SPI2
输出参数	无
返回值	Master Error 标志： 0：未产生 Master Error 1：产生 Master Error

### 22.2.57 SPIx\_SPIIF\_SERR\_Chk

函数名	SPIx_SPIIF_SERR_Chk
函数原型	FlagStatus SPIx_SPIIF_SERR_Chk(SPIx_Type* SPIx)
功能描述	读取 Slave Error 标志
输入参数 1	SPI1/SPI2
输出参数	无
返回值	Slave Error 标志： 0：未产生 Slave Error 1：产生 Slave Error



**22.2.58 SPIx\_SPICR2\_TXO\_AC\_Set**

函数名	SPIx_SPICR2_TXO_AC_Set
函数原型	void SPIx_SPICR2_TXO_AC_Set(SPIx_Type* SPIx, uint32_t SetValue)
功能描述	设置 TXONLY 硬件自动清空的使能
输入参数 1	SPI1/SPI2
输入参数 2	TXONLY 硬件自动清空使能： DISABLE: 禁止 TXONLY 硬件自动清零 ENABLE: TXONLY 硬件自动清零有效，软件使能 TXO 后，等待发送完毕后，硬件清零
输出参数	无
返回值	无

**22.2.59 SPIx\_SPICR2\_TXO\_AC\_Get**

函数名	SPIx_SPICR2_TXO_AC_Get
函数原型	uint32_t SPIx_SPICR2_TXO_AC_Get(SPIx_Type* SPIx)
功能描述	读取 TXONLY 硬件自动清空的使能状态
输入参数 1	SPI1/SPI2
输出参数	无
返回值	TXONLY 硬件自动清空使能： DISABLE: 禁止 TXONLY 硬件自动清零 ENABLE: TXONLY 硬件自动清零有效，软件使能 TXO 后，等待发送完毕后，硬件清零

**22.2.60 SPIx\_SPICR2\_TXO\_Setable**

函数名	SPIx_SPICR2_TXO_Setable
函数原型	void SPIx_SPICR2_TXO_Setable(SPIx_Type* SPIx, FunState NewState)
功能描述	设置 TTXONLY 控制位
输入参数 1	SPI1/SPI2
输入参数 2	TTXONLY 控制位： DISABLE: 禁止单发送模式 ENABLE: 使能 Master 的单发送模式
输出参数	无
返回值	无

**22.2.61 SPIx\_SPICR2\_TXO\_Getable**

函数名	SPIx_SPICR2_TXO_Getable
-----	-------------------------

函数原型	FunState SPIx_SPICR2_TXO_Getable(SPIx_Type* SPIx)
功能描述	读取 TTXONLY 控制位设置
输入参数 1	SPI1/SPI2
输出参数	无
返回值	TTXONLY 控制位： DISABLE：禁止单发送模式 ENABLE：使能 Master 的单发送模式

## 23 HSPI

### 23.1 HSPI 寄存器结构

寄存器	描述
HSPICR1	HSPI 控制寄存器 1
HSPICR2	HSPI 控制寄存器 2
HSPICR3	HSPI 控制寄存器 3
HSPIIE	HSPI 中断控制寄存器
HSPIIF	HSPI 中断标志寄存器
HSPITXBUF	HSPI 发送缓冲寄存器
HSPIRXBUF	HSPI 接收缓冲寄存器

### 23.2 HSPI 库函数

序号	函数名	描述
1	HSPI_Deinit	HSPI 初始化为缺省值
2	HSPI_Master_Init	HSPI 主模式初始化
3	HSPI_Slave_Init	HSPI 从模式初始化
4	HSPI_SPICR2_HSPIEN_Setable	设置 HSPI 使能
5	HSPI_SPICR2_HSPIEN_Getable	读取 HSPI 使能的设置
6	HSPI_SPICR3_TXBFC_Clr	清除发送 BUFFER
7	HSPI_SPICR3_RXBFC_Clr	清除接收 BUFFER
8	HSPI_SPICR3_MERRC_Clr	清除主模式错误
9	HSPI_SPICR3_SERRC_Clr	清除从模式错误
10	HSPI_SPIIE_ERRIE_Setable	设置 HSPI 错误中断使能
11	HSPI_SPIIE_ERRIE_Getable	读取 HSPI 错误中断使能的设置



12	HSPI_SPIIE_TXIE_Setable	设置发送中断使能
13	HSPI_SPIIE_TXIE_Getable	读取发送中断使能的设置
14	HSPI_SPIIE_RXIE_Setable	设置接收中断使能
15	HSPI_SPIIE_RXIE_Getable	读取接收中断使能
16	HSPI_SSN_Set_Low	设置 HSPI 的 SSN 为低电平
17	HSPI_SSN_Set_High	设置 HSPI 的 SSN 为高电平
18	HSPI_Send_Byte	HSPI 接收一个字节
19	HSPI_Recv_Byte	HSPI 发送一个字节
20	HSPI_RW_Byte	HSPI 发送和接收一个字节
21	HSPI_SPITXBUF_Write	写 HSPI 发送缓存
22	HSPI_SPITXBUF_Read	读取 HSPI 发送缓存
23	HSPI_SPIRXBUF_Write	写 HSPI 接收缓存
24	HSPI_SPIRXBUF_Read	读取 HSPI 接收缓存
25	HSPI_SPICR1_CPHOL_Set	设置时钟极性
26	HSPI_SPICR1_CPHOL_Get	读取时钟极性设置
27	HSPI_SPICR1_CPHA_Set	设置时钟相位
28	HSPI_SPICR1_CPHA_Get	读取时钟相位设置
29	HSPI_SPICR1_LSBF_Set	设置帧格式
30	HSPI_SPICR1_LSBF_Get	读取帧格式的设置
31	HSPI_SPICR1_MM_Set	设置 MASTER/SLAVE 模式选择
32	HSPI_SPICR1_MM_Get	读取 MASTER/SLAVE 模式选择
33	HSPI_SPICR1_WAIT_Set	Master 模式下设置加入 SCK cycle 的数值
34	HSPI_SPICR1_WAIT_Get	读取 Master 模式下设置加入 SCK cycle 的数值
35	HSPI_SPICR1_BAUD_Set	MASTER 模式设置波特率
36	HSPI_SPICR1_BAUD_Get	读取 MASTER 模式波特率的设置
37	HSPI_SPICR2_SSNM_Set	Master 模式下 SSN 控制模式的选择
38	HSPI_SPICR2_SSNM_Get	读取 Master 模式下 SSN 控制模式选择
39	HSPI_SPICR2_SSN_Set	Master 模式下设置软件控制 SSN 输出电平
40	HSPI_SPICR2_SSN_Get	读取 Master 模式下软件控制 SSN 输出电平
41	HSPI_SPICR2_SSNSSEN_Set	Master 模式下设置 SSN 的控制方式

42	HSPI_SPICR2_SSSEN_Get	读取 Master 模式下 SSN 的控制方式
43	HSPI_SPICR1_DELAY_CFG_Set	设置 Master 下延迟采样时间或 Slave 下提前发送时间
44	HSPI_SPICR1_DELAY_CFG_Get	获取 Master 下延迟采样时间或 Slave 下提前发送时间
45	HSPI_SPICR2_FLTEN_Setable	设置 Slave 输入管脚滤波使能
46	HSPI_SPICR2_FLTEN_Getable	读取 Slave 输入管脚滤波使能状态
47	HSPI_SPIIF_TXBE_Chk	读取 TX Buffer Empty 标志位
48	HSPI_SPIIF_RXBF_Chk	读取 RX Buffer Full 标志位
49	HSPI_SPIIF_BUSY_Chk	读取 HSPI 空闲标志
50	HSPI_SPIIF_TXCOL_Clr	清除发送缓存溢出标志
51	HSPI_SPIIF_TXCOL_Chk	读取发送缓存溢出标志
52	HSPI_SPIIF_RXCOL_Clr	清除接收缓存溢出标志
53	HSPI_SPIIF_RXCOL_Chk	读取接收缓存溢出标志
54	HSPI_SPIIF_MERR_Chk	读取 Master Error 标志
55	HSPI_SPIIF_SERR_Chk	读取 Slave Error 标志
56	HSPI_SPICR2_TXO_AC_Set	设置 TXONLY 硬件自动清空的使能
57	HSPI_SPICR2_TXO_AC_Get	读取 TXONLY 硬件自动清空的使能状态
58	HSPI_SPICR2_TXO_Setable	设置 TTXONLY 控制位
59	HSPI_SPICR2_TXO_Getable	读取 TTXONLY 控制位设置

### 23.2.1 HSPI\_Deinit

函数名	HSPI_Deinit
函数原型	void HSPI_Deinit(void)
功能描述	HSPI 初始化为缺省值
输入参数 1	无
输出参数	无
返回值	无

### 23.2.2 HSPI\_Master\_Init

函数名	HSPI_Master_Init
函数原型	void HSPI_Master_Init(HSPI_Master_SInitTypeDef* para)
功能描述	HSPI 主模式初始化

输入参数 1	HSPI_Master 参数:
	波特率
	高低位发送顺序: 0: 先发低位 1: 先发高位
	时钟极性: 0: 串行时钟停止在低电平 1: 串行时钟停止在高电平
	时钟相位选择: 0: 第一个时钟边沿时第一个捕捉边沿 1: 第二个时钟边沿时第一个捕捉边沿
	Master 模式下 SSN 控制模式选择: 0: 每次发完 8bit 后 Master 保持 SSN 为低 1: 每次发完 8bit 后 Master 拉高 SSN
	Master 模式下,SSN 的控制源: 0: 硬件自动控制 1: 软件控制
输出参数	无
返回值	无

### 23.2.3 HSPI\_Slave\_Init

函数名	HSPI_Slave_Init
函数原型	void HSPI_Slave_Init(HSPI_Slave_SInitTypeDef* para)
功能描述	HSPI 从模式初始化
输入参数 1	SPI_Slave 参数:
	高低位发送顺序: 0: 先发低位 1: 先发高位
	时钟极性: 0: 串行时钟停止在低电平 1: 串行时钟停止在高电平
	时钟相位选择: 0: 第一个时钟边沿时第一个捕捉边沿 1: 第二个时钟边沿时第一个捕捉边沿
输出参数	无
返回值	无

### 23.2.4 HSPI\_SPICR2\_HSPIEN\_Setable

函数名	HSPI_SPICR2_HSPIEN_Setable
-----	----------------------------

函数原型	void HSPI_SPICR2_HSPIEN_Setable(FunState NewState)
功能描述	设置 HSPI 使能
输入参数 1	DISABLE: 禁止 HSPI, 清空发送接收缓存 ENABLE: 使能 HSPI
输出参数	无
返回值	无

### 23.2.5 HSPI\_SPICR2\_HSPIEN\_Getable

函数名	HSPI_SPICR2_HSPIEN_Getable
函数原型	FunState HSPI_SPICR2_HSPIEN_Getable(void)
功能描述	读取 HSPI 使能的设置
输入参数 1	无
输出参数	无
返回值	DISABLE: 禁止 HSPI, 清空发送接收缓存 ENABLE: 使能 HSPI

### 23.2.6 HSPI\_SPICR3\_TXBFC\_Clr

函数名	HSPI_SPICR3_TXBFC_Clr
函数原型	void HSPI_SPICR3_TXBFC_Clr(void)
功能描述	清除发送 BUFFER
输入参数 1	无
输出参数	无
返回值	无

### 23.2.7 HSPI\_SPICR3\_RXBFC\_Clr

函数名	HSPI_SPICR3_RXBFC_Clr
函数原型	void HSPI_SPICR3_RXBFC_Clr(void)
功能描述	清除接收 BUFFER
输入参数 1	无
输出参数	无
返回值	无

### 23.2.8 HSPI\_SPICR3\_MERRC\_Clr

函数名	HSPI_SPICR3_MERRC_Clr
-----	-----------------------

函数原型	void HSPI_SPICR3_MERRC_Clr(void)
功能描述	清除主模式错误
输入参数 1	无
输出参数	无
返回值	无

### 23.2.9 HSPI\_SPICR3\_SERRC\_Clr

函数名	HSPI_SPICR3_SERRC_Clr
函数原型	void HSPI_SPICR3_SERRC_Clr(void)
功能描述	清除从模式错误
输入参数 1	无
输出参数	无
返回值	无

### 23.2.10 HSPI\_SPIIE\_ERRIE\_Setable

函数名	HSPI_SPIIE_ERRIE_Setable
函数原型	void HSPI_SPIIE_ERRIE_Setable(FunState NewState)
功能描述	设置 HSPI 错误中断使能
输入参数 1	HSPI 错误中断使能： DISABLE：禁止 HSPI 错误中断 ENABLE：使能 HSPI 错误中断
输出参数	无
返回值	无

### 23.2.11 HSPI\_SPIIE\_ERRIE\_Getable

函数名	HSPI_SPIIE_ERRIE_Getable
函数原型	FunState HSPI_SPIIE_ERRIE_Getable(void)
功能描述	读取 HSPI 错误中断使能的设置
输入参数 1	无
输出参数	无
返回值	HSPI 错误中断使能： DISABLE：禁止 HSPI 错误中断 ENABLE：使能 HSPI 错误中断

**23.2.12 HSPI\_SPIIE\_TXIE\_Setable**

函数名	HSPI_SPIIE_TXIE_Setable
函数原型	void HSPI_SPIIE_TXIE_Setable(FunState NewState)
功能描述	设置发送中断使能
输入参数 1	发送中断使能： DISABLE：禁止发送中断 ENABLE：使能发送中断
输出参数	无
返回值	无

**23.2.13 HSPI\_SPIIE\_TXIE\_Getable**

函数名	HSPI_SPIIE_TXIE_Getable
函数原型	FunState HSPI_SPIIE_TXIE_Getable(void)
功能描述	读取发送中断使能的设置
输入参数 1	无
输出参数	无
返回值	发送中断使能： DISABLE：禁止发送中断 ENABLE：使能发送中断

**23.2.14 HSPI\_SPIIE\_RXIE\_Setable**

函数名	HSPI_SPIIE_RXIE_Setable
函数原型	void HSPI_SPIIE_RXIE_Setable(FunState NewState)
功能描述	设置接收中断使能
输入参数 1	接收中断使能： DISABLE：禁止接收中断 ENABLE：使能接收中断
输出参数	无
返回值	无

**23.2.15 HSPI\_SPIIE\_RXIE\_Getable**

函数名	HSPI_SPIIE_RXIE_Getable
函数原型	FunState HSPI_SPIIE_RXIE_Getable(void)
功能描述	读取接收中断使能



输入参数 1	无
输出参数	无
返回值	接收中断使能： DISABLE：禁止发送中断 ENABLE：使能发送中断

### 23.2.16 HSPI\_SSN\_Set\_Low

函数名	HSPI_SSN_Set_Low
函数原型	void HSPI_SSN_Set_Low(void)
功能描述	设置 HSPI 的 SSN 为低电平
输入参数 1	无
输出参数	无
返回值	无

### 23.2.17 HSPI\_SSN\_Set\_High

函数名	HSPI_SSN_Set_High
函数原型	HSPI_SSN_Set_High(void)
功能描述	设置 HSPI 的 SSN 为高电平
输入参数 1	无
输出参数	无
返回值	无

### 23.2.18 HSPI\_Send\_Byte

函数名	HSPI_Send_Byte
函数原型	void HSPI_Send_Byte(unsigned char data)
功能描述	HSPI 接收一个字节
输入参数 1	待发送的数据:0-FF
输出参数	无
返回值	无

### 23.2.19 HSPI\_Rcv\_Byte

函数名	HSPI_Rcv_Byte
函数原型	unsigned char HSPI_Rcv_Byte(void)
功能描述	HSPI 发送一个字节

输入参数 1	无
输出参数	无
返回值	接收的数据:0-FF

### 23.2.20 HSPI\_RW\_Byte

函数名	HSPI_RW_Byte
函数原型	unsigned char HSPI_RW_Byte(unsigned char data)
功能描述	HSPI 发送和接收一个字节
输入参数 1	待发送的数据:0-FF
输出参数	无
返回值	接收的数据: 0-FF

### 23.2.21 HSPI\_SPITXBUF\_Write

函数名	HSPI_SPITXBUF_Write
函数原型	void HSPI_SPITXBUF_Write(uint32_t SetValue)
功能描述	写 HSPI 发送缓存
输入参数 1	HSPI 发送缓存数据:0-FF
输出参数	无
返回值	无

### 23.2.22 HSPI\_SPITXBUF\_Read

函数名	HSPI_SPITXBUF_Read
函数原型	uint32_t HSPI_SPITXBUF_Read(void)
功能描述	读取 HSPI 发送缓存
输入参数 1	无
输出参数	无
返回值	HSPI 发送缓存的值:0-FF

### 23.2.23 HSPI\_SPIRXBUF\_Write

函数名	HSPI_SPIRXBUF_Write
函数原型	void HSPI_SPIRXBUF_Write(uint32_t SetValue)
功能描述	写 HSPI 接收缓存
输入参数 1	HSPI 接收缓存数据: 0-FF
输出参数	无

返回值	无
-----	---

### 23.2.24 HSPI\_SPIRXBUF\_Read

函数名	HSPI_SPIRXBUF_Read
函数原型	uint32_t HSPI_SPIRXBUF_Read(void)
功能描述	读取 HSPI 接收缓存
输入参数 1	无
输出参数	无
返回值	HSPI 接收缓存的值:0-FF

### 23.2.25 HSPI\_SPICR1\_CPHOL\_Set

函数名	HSPI_SPICR1_CPHOL_Set
函数原型	void HSPI_SPICR1_CPHOL_Set(uint32_t SetValue)
功能描述	设置时钟极性
输入参数 1	时钟极性: 0: 串行时钟停止在低电平 1: 串行时钟停止在高电平
输出参数	无
返回值	无

### 23.2.26 HSPI\_SPICR1\_CPHOL\_Get

函数名	HSPI_SPICR1_CPHOL_Get
函数原型	uint32_t HSPI_SPICR1_CPHOL_Get(void)
功能描述	读取时钟极性设置
输入参数 1	无
输出参数	无
返回值	时钟极性: 0: 串行时钟停止在低电平 1: 串行时钟停止在高电平

### 23.2.27 HSPI\_SPICR1\_CPHA\_Set

函数名	HSPI_SPICR1_CPHA_Set
函数原型	void HSPI_SPICR1_CPHA_Set(uint32_t SetValue)
功能描述	设置时钟相位

输入参数 1	时钟相位选择： 0：第一个时钟边沿时第一个捕捉边沿 1：第二个时钟边沿时第一个捕捉边沿
输出参数	无
返回值	无

### 23.2.28 HSPI\_SPICR1\_CPHA\_Get

函数名	HSPI_SPICR1_CPHA_Get
函数原型	uint32_t HSPI_SPICR1_CPHA_Get(void)
功能描述	读取时钟相位设置
输入参数 1	无
输出参数	无
返回值	时钟相位选择： 0：第一个时钟边沿时第一个捕捉边沿 1：第二个时钟边沿时第一个捕捉边沿

### 23.2.29 HSPI\_SPICR1\_LSBF\_Set

函数名	HSPI_SPICR1_LSBF_Set
函数原型	void HSPI_SPICR1_LSBF_Set(uint32_t SetValue)
功能描述	设置帧格式
输入参数 1	时钟相位选择： 0：第一个时钟边沿时第一个捕捉边沿 1：第二个时钟边沿时第一个捕捉边沿
输出参数	无
返回值	无

### 23.2.30 HSPI\_SPICR1\_LSBF\_Get

函数名	HSPI_SPICR1_LSBF_Get
函数原型	uint32_t HSPI_SPICR1_LSBF_Get(void)
功能描述	读取帧格式的设置
输入参数 1	无
输出参数	无
返回值	时钟相位选择： 0：第一个时钟边沿时第一个捕捉边沿 1：第二个时钟边沿时第一个捕捉边沿

**23.2.31 HSPI\_SPICR1\_MM\_Set**

函数名	HSPI_SPICR1_MM_Set
函数原型	void HSPI_SPICR1_MM_Set(uint32_t SetValue)
功能描述	设置 MASTER/SLAVE 模式选择
输入参数 1	MASTER/SLAVE 模式选择: 0: Slave 模式 1: Master 模式
输出参数	无
返回值	无

**23.2.32 HSPI\_SPICR1\_MM\_Get**

函数名	HSPI_SPICR1_MM_Get
函数原型	uint32_t HSPI_SPICR1_MM_Get(void)
功能描述	读取 MASTER/SLAVE 模式选择
输入参数 1	无
输出参数	无
返回值	MASTER/SLAVE 模式选择: 0: Slave 模式 1: Master 模式

**23.2.33 HSPI\_SPICR1\_WAIT\_Set**

函数名	HSPI_SPICR1_WAIT_Set
函数原型	void HSPI_SPICR1_WAIT_Set(uint32_t SetValue)
功能描述	Master 模式下设置发送完每个 byte 之后的等待时间
输入参数 1	发送完每个 byte 之后的等待时间: HSPI_SPICR1_WAIT_WAIT_1: 每发完 8bit 至少加入 1 个 sck 周期等待时间再传输下一个 8Bit 数据 HSPI_SPICR1_WAIT_WAIT_2 每发完 8bit 至少加入 2 个 sck 周期等待时间再传输下一个 8Bit 数据 HSPI_SPICR1_WAIT_WAIT_3: 每发完 8bit 至少加入 3 个 sck 周期等待时间再传输下一个 8Bit 数据 HSPI_SPICR1_WAIT_WAIT_4: 每发完 8bit 至少加入 4 个 sck 周期等待时间再传输下一个 8Bit 数据
输出参数	无

返回值	无
-----	---

### 23.2.34 HSPI\_SPICR1\_WAIT\_Get

函数名	HSPI_SPICR1_WAIT_Get
函数原型	uint32_t HSPI_SPICR1_WAIT_Get(void)
功能描述	读取 Master 模式下设置发送完每个 byte 之后的等待时间
输入参数 1	无
输出参数	无
返回值	发送完每个 byte 之后的等待时间： HSPI_SPICR1_WAIT_WAIT_1: 每发完 8bit 至少加入 1 个 sck 周期等待时间再传输下一个 8Bit 数据 HSPI_SPICR1_WAIT_WAIT_2 每发完 8bit 至少加入 2 个 sck 周期等待时间再传输下一个 8Bit 数据 HSPI_SPICR1_WAIT_WAIT_3: 每发完 8bit 至少加入 3 个 sck 周期等待时间再传输下一个 8Bit 数据 HSPI_SPICR1_WAIT_WAIT_4: 每发完 8bit 至少加入 4 个 sck 周期等待时间再传输下一个 8Bit 数据

### 23.2.35 HSPI\_SPICR1\_BAUD\_Set

函数名	HSPI_SPICR1_BAUD_Set
函数原型	void HSPI_SPICR1_BAUD_Set(uint32_t SetValue)
功能描述	MASTER 模式设置波特率分频值
输入参数 1	MASTER 模式波特率分频值： 000: fAHBCLK 001: fAHBCLK/2 010: fAHBCLK/4 011: fAHBCLK/8 100: fAHBCLK/16 101: fAHBCLK/32 110: fAHBCLK/64 111: fAHBCLK/128
输出参数	无
返回值	无

**23.2.36 HSPI\_SPICR1\_BAUD\_Get**

函数名	HSPI_SPICR1_BAUD_Get
函数原型	uint32_t HSPI_SPICR1_BAUD_Get(void)
功能描述	读取 MASTER 模式波特率分频值
输入参数 1	无
输出参数	无
返回值	MASTER 模式波特率分频值： 000: fAHBCLK 001: fAHBCLK/2 010: fAHBCLK/4 011: fAHBCLK/8 100: fAHBCLK/16 101: fAHBCLK/32 110: fAHBCLK/64 111: fAHBCLK/128

**23.2.37 HSPI\_SPICR2\_SSNM\_Set**

函数名	HSPI_SPICR2_SSNM_Set
函数原型	void HSPI_SPICR2_SSNM_Set(uint32_t SetValue)
功能描述	Master 模式下 SSN 控制模式的选择
输入参数 1	Master 模式下 SSN 控制模式选择： 0: 每次发完 8bit 后 Master 保持 SSN 为低 1: 每次发完 8bit 后 Master 拉高 SSN
输出参数	无
返回值	无

**23.2.38 HSPI\_SPICR2\_SSNM\_Get**

函数名	HSPI_SPICR2_SSNM_Get
函数原型	uint32_t HSPI_SPICR2_SSNM_Get(void)
功能描述	读取 Master 模式下 SSN 控制模式选择
输入参数 1	无
输出参数	无
返回值	Master 模式下 SSN 控制模式选择： 0: 每次发完 8bit 后 Master 保持 SSN 为低 1: 每次发完 8bit 后 Master 拉高 SSN

**23.2.39 HSPI\_SPICR2\_SSN\_Set**

函数名	HSPI_SPICR2_SSN_Set
函数原型	void HSPI_SPICR2_SSN_Set(uint32_t SetValue)
功能描述	Master 模式下设置软件控制 SSN 输出电平
输入参数 1	Master 模式下软件控制 SSN 输出电平： 0: SSN 输出低电平 1: SSN 输出高电平
输出参数	无
返回值	无

**23.2.40 HSPI\_SPICR2\_SSN\_Get**

函数名	HSPI_SPICR2_SSN_Get
函数原型	uint32_t HSPI_SPICR2_SSN_Get(void)
功能描述	读取 Master 模式下软件控制 SSN 输出电平
输入参数 1	无
输出参数	无
返回值	Master 模式下软件控制 SSN 输出电平： 0: SSN 输出低电平 1: SSN 输出高电平

**23.2.41 HSPI\_SPICR2\_SSNSSEN\_Set**

函数名	HSPI_SPICR2_SSNSSEN_Set
函数原型	void HSPI_SPICR2_SSNSSEN_Set(uint32_t SetValue)
功能描述	Master 模式下设置 SSN 的控制方式
输入参数 1	Master 模式下,SSN 的控制模式： 0: 硬件自动控制 1: 软件控制
输出参数	无
返回值	无

**23.2.42 HSPI\_SPICR2\_SSNSSEN\_Get**

函数名	HSPI_SPICR2_SSNSSEN_Get
函数原型	uint32_t HSPI_SPICR2_SSNSSEN_Get(void)



功能描述	读取 Master 模式下 SSN 的控制方式
输入参数 1	无
输出参数	无
返回值	Master 模式下,SSN 的控制模式: 0: 硬件自动控制 1: 软件控制

### 23.2.43 HSPI\_SPICR1\_DELAY\_CFG\_Set

函数名	HSPI_SPICR1_DELAY_CFG_Set
函数原型	void HSPI_SPICR1_DELAY_CFG_Set(uint32_t SetValue)
功能描述	设置 Master 下延迟采样时间或 Slave 下提前发送时间
输入参数 1	Master 下延迟采样时间或 Slave 下提前发送时间: HSPI_SPICR1_DELAY_CFG_MASTER_NO_DELAY: Master 模式, 无延迟 HSPI_SPICR1_DELAY_CFG_SLAVE_NORMAL: Slave 模式, 正常发送 HSPI_SPICR1_DELAY_CFG_MASTER_DLY_L1: Master 模式, 1 级延迟 HSPI_SPICR1_DELAY_CFG_SLAVE_PHASE_1: Slave 模式, 发送相位 1 HSPI_SPICR1_DELAY_CFG_MASTER_DLY_L2: Master 模式, 2 级延迟 HSPI_SPICR1_DELAY_CFG_SLAVE_PHASE_2: Slave 模式, 发送相位 2 HSPI_SPICR1_DELAY_CFG_MASTER_DLY_L3: Master 模式, 3 级延迟 HSPI_SPICR1_DELAY_CFG_SLAVE_PHASE_3: Slave 模式, 发送相位 3 HSPI_SPICR1_DELAY_CFG_MASTER_DLY_L4: Master 模式, 4 级延迟 HSPI_SPICR1_DELAY_CFG_SLAVE_PHASE_4: Slave 模式, 发送相位 4
输出参数	无
返回值	无

### 23.2.44 HSPI\_SPICR1\_DELAY\_CFG\_Get

函数名	HSPI_SPICR1_DELAY_CFG_Get
函数原型	uint32_t HSPI_SPICR1_DELAY_CFG_Get(void)
功能描述	获取 Master 下延迟采样时间或 Slave 下提前发送时间
输入参数 1	无
输出参数	无
返回值	Master 下延迟采样时间或 Slave 下提前发送时间: HSPI_SPICR1_DELAY_CFG_MASTER_NO_DELAY: Master 模式, 无延迟 HSPI_SPICR1_DELAY_CFG_SLAVE_NORMAL: Slave 模式, 正常发送 HSPI_SPICR1_DELAY_CFG_MASTER_DLY_L1: Master 模式, 1 级延迟 HSPI_SPICR1_DELAY_CFG_SLAVE_PHASE_1: Slave 模式, 发送相位 1

	HSPI_SPICR1_DELAY_CFG_MASTER_DLY_L2: Master 模式, 2 级延迟 HSPI_SPICR1_DELAY_CFG_SLAVE_PHASE_2: Slave 模式, 发送相位 2 HSPI_SPICR1_DELAY_CFG_MASTER_DLY_L3: Master 模式, 3 级延迟 HSPI_SPICR1_DELAY_CFG_SLAVE_PHASE_3: Slave 模式, 发送相位 3 HSPI_SPICR1_DELAY_CFG_MASTER_DLY_L4: Master 模式, 4 级延迟 HSPI_SPICR1_DELAY_CFG_SLAVE_PHASE_4: Slave 模式, 发送相位 4
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### 23.2.45 HSPI\_SPICR2\_FLTEN\_Setable

函数名	HSPI_SPICR2_FLTEN_Setable
函数原型	void HSPI_SPICR2_FLTEN_Setable(FunState NewState)
功能描述	设置 Slave 输入管脚滤波使能
输入参数 1	Slave 输入管脚滤波使能: DISABLE: 不滤波 ENABLE: 使能 4ns 滤波
输出参数	无
返回值	无

### 23.2.46 HSPI\_SPICR2\_FLTEN\_Getable

函数名	HSPI_SPICR2_FLTEN_Getable
函数原型	FunState HSPI_SPICR2_FLTEN_Getable(void)
功能描述	读取 Slave 输入管脚滤波使能状态
输入参数 1	无
输出参数	无
返回值	Slave 输入管脚滤波使能: DISABLE: 不滤波 ENABLE: 使能 4ns 滤波

### 23.2.47 HSPI\_SPIIF\_TXBE\_Chk

函数名	HSPI_SPIIF_TXBE_Chk
函数原型	FlagStatus HSPI_SPIIF_TXBE_Chk(void)
功能描述	读取 TX Buffer Empty 标志位
输入参数 1	无
输出参数	无
返回值	TX Buffer Empty 标志位: 0: 发送缓存满

1: 发送缓存空

**23.2.48 HSPI\_SPIIF\_RXBF\_Chk**

函数名	HSPI_SPIIF_RXBF_Chk
函数原型	FlagStatus HSPI_SPIIF_RXBF_Chk(void)
功能描述	读取 RX Buffer Full 标志位
输入参数 1	无
输出参数	无
返回值	RX Buffer Full 标志位: 0: 接收缓存空 1: 接收缓存满

**23.2.49 HSPI\_SPIIF\_BUSY\_Chk**

函数名	HSPI_SPIIF_BUSY_Chk
函数原型	FlagStatus HSPI_SPIIF_BUSY_Chk(void)
功能描述	读取 HSPI 空闲标志
输入参数 1	无
输出参数	无
返回值	SPI 空闲标志: 0: SPI 忙 1: SPI 空闲

**23.2.50 HSPI\_SPIIF\_TXCOL\_Clr**

函数名	HSPI_SPIIF_TXCOL_Clr
函数原型	void HSPI_SPIIF_TXCOL_Clr(void)
功能描述	清除发送缓存溢出标志
输入参数 1	无
输出参数	无
返回值	无

**23.2.51 HSPI\_SPIIF\_TXCOL\_Chk**

函数名	HSPI_SPIIF_TXCOL_Chk
函数原型	FlagStatus HSPI_SPIIF_TXCOL_Chk(void)
功能描述	读取发送缓存溢出标志



输入参数 1	无
输出参数	无
返回值	发送缓存溢出标志： 0：发送缓存未溢出 1：发送缓存溢出

### 23.2.52 HSPI\_SPIIF\_RXCOL\_Clr

函数名	HSPI_SPIIF_RXCOL_Clr
函数原型	void HSPI_SPIIF_RXCOL_Clr(void)
功能描述	清除接收缓存溢出标志
输入参数 1	无
输出参数	无
返回值	无

### 23.2.53 HSPI\_SPIIF\_RXCOL\_Chk

函数名	HSPI_SPIIF_RXCOL_Chk
函数原型	FlagStatus HSPI_SPIIF_RXCOL_Chk(void)
功能描述	读取接收缓存溢出标志
输入参数 1	无
输出参数	无
返回值	接收缓存溢出标志： 0：接收缓存未溢出 1：接收缓存溢出

### 23.2.54 HSPI\_SPIIF\_MERR\_Chk

函数名	HSPI_SPIIF_MERR_Chk
函数原型	FlagStatus HSPI_SPIIF_MERR_Chk(void)
功能描述	读取 Master Error 标志
输入参数 1	无
输出参数	无
返回值	Master Error 标志： 0：未产生 Master Error 1：产生 Master Error

**23.2.55 HSPI\_SPIIF\_SERR\_Chk**

函数名	HSPI_SPIIF_SERR_Chk
函数原型	FlagStatus HSPI_SPIIF_SERR_Chk(void)
功能描述	读取 Slave Error 标志
输入参数 1	无
输出参数	无
返回值	Slave Error 标志: 0: 未产生 Slave Error 1: 产生 Slave Error

**23.2.56 HSPI\_SPICR2\_TXO\_AC\_Set**

函数名	HSPI_SPICR2_TXO_AC_Set
函数原型	HSPI_SPICR2_TXO_AC_Set(uint32_t SetValue)
功能描述	设置 TXONLY 硬件自动清零的使能
输入参数 1	TXONLY 硬件自动清零使能: DISABLE: 禁止 TXONLY 硬件自动清零 ENABLE: TXONLY 硬件自动清零有效, 软件使能 TXO 后, 等待发送完毕后, 硬件清零
输出参数	无
返回值	无

**23.2.57 HSPI\_SPICR2\_TXO\_AC\_Get**

函数名	HSPI_SPICR2_TXO_AC_Get
函数原型	uint32_t HSPI_SPICR2_TXO_AC_Get(void)
功能描述	读取 TXONLY 硬件自动清零的使能状态
输入参数 1	无
输出参数	无
返回值	TXONLY 硬件自动清零使能: DISABLE: 禁止 TXONLY 硬件自动清零 ENABLE: TXONLY 硬件自动清零有效, 软件使能 TXO 后, 等待发送完毕后, 硬件清零

**23.2.58 HSPI\_SPICR2\_TXO\_Setable**

函数名	HSPI_SPICR2_TXO_Setable
函数原型	HSPI_SPICR2_TXO_Setable(FunState NewState)
功能描述	设置 TTXONLY 控制位



输入参数 1	TTXONLY 控制位： DISABLE：禁止单发送模式 ENABLE：使能 Master 的单发送模式
输出参数	无
返回值	无

### 23.2.59 HSPI\_SPICR2\_TXO\_Getable

函数名	HSPI_SPICR2_TXO_Getable
函数原型	FunState HSPI_SPICR2_TXO_Getable(void)
功能描述	读取 TTXONLY 控制位设置
输入参数 1	无
输出参数	无
返回值	TTXONLY 控制位： DISABLE：禁止单发送模式 ENABLE：使能 Master 的单发送模式



## 24 版本说明

版本	日志
V1.0	首次发布