持ち込み可能なヘッダファイルの例(今回の課題に特化していない汎用的な設定などが記述されています。)

```
/***********************************************************************/
/*                                                                     */
/*  FILE        :sfr_r835a.h                                           */
/*  DATE        :Tue, Mar 17, 2009                                     */
/*  DESCRIPTION :define the sfr register. (for C language)             */
/*  CPU GROUP   :35A                                                   */
/*                                                                     */
/*  This file is generated by Renesas Project Generator (Ver.4.12).    */
/*                                                                     */
/***********************************************************************/

/************************************************************************************
*                                                                                   *
*  File Name    : SFR_R835a.h                                                       *
*  Contents     : definition of R8C/35A Group SFR                                   *
*  Copyright, 2007 RENESAS TECHNOLOGY CORPORATION                                   *
*  Version      : 1.00 (07-09-25)                                                   *
*                                                                                   *
*  note         :                                                                   *
*                                                                                   *
************************************************************************************/

/*********************************************************
*  declare SFR addresses                                 *
*********************************************************/

#pragma ADDRESS pm0_addr 0004H /* Processor mode register 0 */
#pragma ADDRESS pm1_addr 0005H /* Processor mode register 1 */
#pragma ADDRESS cm0_addr 0006H /* System clock control register 0 */
#pragma ADDRESS cm1_addr 0007H /* System clock control register 1 */
#pragma ADDRESS mstcr_addr 0008H /* Module clock enable register */
#pragma ADDRESS cm3_addr 0009H /* System clock control register 3 *//* 3x */
#pragma ADDRESS prcr_addr 000AH /* Protect register */
#pragma ADDRESS rstfr_addr 000BH /* Reset factor register *//* 3x */
#pragma ADDRESS ocd_addr 000CH /* Oscillation stop detect register */
#pragma ADDRESS wdtr_addr 000DH /* Watchdog timer reset register */
#pragma ADDRESS wdts_addr 000EH /* Watchdog timer start register */
#pragma ADDRESS wdtc_addr 000FH /* Watchdog timer control register */
#pragma ADDRESS fra7_addr 0015H /* 40MHz onchip oscillator A control register 7 */
#pragma ADDRESS cspr_addr 001CH /* Count source protect mode register */
#pragma ADDRESS fra0_addr 0023H /* 40MHz onchip oscillator A control register 0 */
#pragma ADDRESS fra1_addr 0024H /* 40MHz onchip oscillator A control register 1 */
#pragma ADDRESS fra2_addr 0025H /* 40MHz onchip oscillator A control register 2 */
#pragma ADDRESS ocvrefcr_addr 0026H /* Internal voltage control register *//* 3x */
#pragma ADDRESS cpsrf_addr 0028H /* Clock prescaler reset flag */
#pragma ADDRESS fra4_addr 0029H /* 40MHz onchip oscillator A control register 4 *//* 3x */
#pragma ADDRESS fra5_addr 002AH /* 40MHz onchip oscillator A control register 5 *//* 3x */
#pragma ADDRESS fra6_addr 002BH /* 40MHz onchip oscillator A control register 6 *//* 3x */
#pragma ADDRESS fra3_addr 002FH /* 40MHz onchip oscillator A control register 3 *//* 3x */
#pragma ADDRESS cmpa_addr 0030H /* Voltage monitor circuit / comparator A control register *//* 3x */
#pragma ADDRESS vcac_addr 0031H /* Voltage monitor circuit edge select register *//* 3x */
#pragma ADDRESS vca1_addr 0033H /* Voltage detection A register 1 *//* 3x */
#pragma ADDRESS vca2_addr 0034H /* Voltage detection A register 2 *//* 3x */
#pragma ADDRESS vd1ls_addr 0036H /* Voltage detection 1 level select register *//* 3x */
#pragma ADDRESS vw0c_addr 0038H /* Voltage monitor 0 circuit control register */
#pragma ADDRESS vw1c_addr 0039H /* Voltage monitor 1 circuit control register *//* 3x */
#pragma ADDRESS vw2c_addr 003AH /* Voltage monitor 2 circuit control register *//* 3x */
#pragma ADDRESS fmrdyic_addr 0041H /* Flash memory ready interupt control register *//* 3x */
#pragma ADDRESS int4ic_addr 0046H /* INT4 interrupt control register *//* 3x */
#pragma ADDRESS trcic_addr 0047H /* Timer RC interrupt control register */
#pragma ADDRESS trd0ic_addr 0048H /* Timer RD0 interrupt control register */
#pragma ADDRESS trd1ic_addr 0049H /* Timer RD1 interrupt control register */
#pragma ADDRESS treic_addr 004AH /* Timer RE interrupt control register */
#pragma ADDRESS s2tic_addr 004BH /* UART2 transmit interrupt control register */
#pragma ADDRESS s2ric_addr 004CH /* UART2 receive interrupt control register */
#pragma ADDRESS kupic_addr 004DH /* Key input interrupt control register */
#pragma ADDRESS adic_addr 004EH /* A-D conversion interrupt control register */
```

```
#pragma ADDRESS ssuic_addr 004FH /* SSU interrupt control register */
#pragma ADDRESS iicic_addr 004FH /* IIC interrupt control register */
#pragma ADDRESS s0tic_addr 0051H /* UART0 transmit interrupt control register */
#pragma ADDRESS s0ric_addr 0052H /* UART0 receive interrupt control register */
#pragma ADDRESS s1tic_addr 0053H /* UART1 transmit interrupt control register */
#pragma ADDRESS s1ric_addr 0054H /* UART1 receive interrupt control register */
#pragma ADDRESS int2ic_addr 0055H /* INT2 interrupt control register */
#pragma ADDRESS traic_addr 0056H /* Timer RA interrupt control register */
#pragma ADDRESS trbic_addr 0058H /* Timer RB interrupt control register */
#pragma ADDRESS int1ic_addr 0059H /* INT1 interrupt control register */
#pragma ADDRESS int3ic_addr 005AH /* INT3 interrupt control register */
#pragma ADDRESS int0ic_addr 005DH /* INT0 interrupt control register */
#pragma ADDRESS u2bcnic_addr 005EH /* UART2 bus collision detection interrupt control register *//* 3x */
#pragma ADDRESS vcmp1ic_addr 0072H /* comparator A1 interrupt control register *//* 3x */
#pragma ADDRESS vcmp2ic_addr 0073H /* comparator A2 interrupt control register *//* 3x */
#pragma ADDRESS dtctl_addr 0080H /* DTC start-up control register *//* 3x */
#pragma ADDRESS dtcen0_addr 0088H /* DTC start-up enable register 0 *//* 3x */
#pragma ADDRESS dtcen1_addr 0089H /* DTC start-up enable register 1 *//* 3x */
#pragma ADDRESS dtcen2_addr 008AH /* DTC start-up enable register 2 *//* 3x */
#pragma ADDRESS dtcen3_addr 008BH /* DTC start-up enable register 3 *//* 3x */
#pragma ADDRESS dtcen4_addr 008CH /* DTC start-up enable register 4 *//* 3x */
#pragma ADDRESS dtcen5_addr 008DH /* DTC start-up enable register 5 *//* 3x */
#pragma ADDRESS dtcen6_addr 008EH /* DTC start-up enable register 6 *//* 3x */
#pragma ADDRESS u0mr_addr 00A0H /* UART0 transmit/receive mode register */
#pragma ADDRESS u0brg_addr 00A1H /* UART0 bit rate register */
#pragma ADDRESS u0tb_addr 00A2H /* UART0 transmit buffer register */
#pragma ADDRESS u0c0_addr 00A4H /* UART0 transmit/receive control register 0 */
#pragma ADDRESS u0c1_addr 00A5H /* UART0 transmit/receive control register 1 */
#pragma ADDRESS u0rb_addr 00A6H /* UART0 receive buffer register */
#pragma ADDRESS u2mr_addr 00A8H /* UART2 transmit/receive mode register *//* 3x */
#pragma ADDRESS u2brg_addr 00A9H /* UART2 bit rate register *//* 3x */
#pragma ADDRESS u2tb_addr 00AAH /* UART2 transmit buffer register *//* 3x */
#pragma ADDRESS u2c0_addr 00ACH /* UART2 transmit/receive control register 0 *//* 3x */
#pragma ADDRESS u2c1_addr 00ADH /* UART2 transmit/receive control register 1 *//* 3x */
#pragma ADDRESS u2rb_addr 00AEH /* UART2 receive buffer register *//* 3x */
#pragma ADDRESS urxdf_addr 00B0H /* UART2 degital filter function select register *//* 3x */
#pragma ADDRESS u2smr5_addr 00BBH /* UART2 special mode register 5 *//* 3x */
#pragma ADDRESS u2smr4_addr 00BCH /* UART2 special mode register 4 *//* 3x */
#pragma ADDRESS u2smr3_addr 00BDH /* UART2 special mode register 3 *//* 3x */
#pragma ADDRESS u2smr2_addr 00BEH /* UART2 special mode register 2 *//* 3x */
#pragma ADDRESS u2smr_addr 00BFH /* UART2 special mode register *//* 3x */
#pragma ADDRESS ad0_addr 00C0H /* A-D register 0 *//* 3x */
#pragma ADDRESS ad1_addr 00C2H /* A-D register 1 *//* 3x */
#pragma ADDRESS ad2_addr 00C4H /* A-D register 2 *//* 3x */
#pragma ADDRESS ad3_addr 00C6H /* A-D register 3 *//* 3x */
#pragma ADDRESS ad4_addr 00C8H /* A-D register 4 *//* 3x */
#pragma ADDRESS ad5_addr 00CAH /* A-D register 5 *//* 3x */
#pragma ADDRESS ad6_addr 00CCH /* A-D register 6 *//* 3x */
#pragma ADDRESS ad7_addr 00CEH /* A-D register 7 *//* 3x */
#pragma ADDRESS admod_addr 00D4H /* A-D mode register *//* 3x */
#pragma ADDRESS adinsel_addr 00D5H /* A-D input select register */
#pragma ADDRESS adcon0_addr 00D6H /* A-D control register 0 */
#pragma ADDRESS adcon1_addr 00D7H /* A-D control register 1 */
#pragma ADDRESS da0_addr 00D8H /* D-A register 0 */
#pragma ADDRESS da1_addr 00D9H /* D-A register 1 *//* 3x */
#pragma ADDRESS dacon_addr 00DCH /* D-A control register */
#pragma ADDRESS p0_addr 00E0H /* Port P0 register */
```

```
#pragma ADDRESS p1_addr 00E1H /* Port P1 register */
#pragma ADDRESS pd0_addr 00E2H /* Port P0 direction register */
#pragma ADDRESS pd1_addr 00E3H /* Port P1 direction register */
#pragma ADDRESS p2_addr 00E4H /* Port P2 register */
#pragma ADDRESS p3_addr 00E5H /* Port P3 register */
#pragma ADDRESS pd2_addr 00E6H /* Port P2 direction register */
#pragma ADDRESS pd3_addr 00E7H /* Port P3 direction register */
#pragma ADDRESS p4_addr 00E8H /* Port P4 register */
#pragma ADDRESS p5_addr 00E9H /* Port P5 register */
#pragma ADDRESS pd4_addr 00EAH /* Port P4 direction register */
#pragma ADDRESS pd5_addr 00EBH /* Port P5 direction register */
#pragma ADDRESS p6_addr 00ECH /* Port P6 register */
#pragma ADDRESS pd6_addr 00EEH /* Port P6 direction register */
#pragma ADDRESS tracr_addr 0100H /* Timer RA control register */
#pragma ADDRESS traioc_addr 0101H /* Timer RA I/O control register */
#pragma ADDRESS tramr_addr 0102H /* Timer RA mode register */
#pragma ADDRESS trapre_addr 0103H /* Timer RA prescaler */
#pragma ADDRESS tra_addr 0104H /* Timer RA */
#pragma ADDRESS lincr2_addr 0105H /* LIN control register 2 */
#pragma ADDRESS lincr_addr 0106H /* LIN control register */
#pragma ADDRESS linst_addr 0107H /* LIN status register */
#pragma ADDRESS trbcr_addr 0108H /* Timer RB control register */
#pragma ADDRESS trbocr_addr 0109H /* Timer RB one-shot control register */
#pragma ADDRESS trbioc_addr 010AH /* Timer RB I/O control register */
#pragma ADDRESS trbmr_addr 010BH /* Timer RB mode register */
#pragma ADDRESS trbpre_addr 010CH /* Timer RB prescaler */
#pragma ADDRESS trbsc_addr 010DH /* Timer RB secondary */
#pragma ADDRESS trbpr_addr 010EH /* Timer RB primary */
#pragma ADDRESS tresec_addr 0118H /* Timer RE seconds data register / Timer RE
counter data register */
#pragma ADDRESS tremin_addr 0119H /* Timer RE minutes data register / Timer
RE compare data register */
#pragma ADDRESS trehr_addr 011AH /* Timer RE hours data register */
#pragma ADDRESS trewk_addr 011BH /* Timer RE days of week data register */
#pragma ADDRESS trecr1_addr 011CH /* Timer RE control register1 */
#pragma ADDRESS trecr2_addr 011DH /* Timer RE control register2 */
#pragma ADDRESS trecsr_addr 011EH /* Timer RE count source select register */
#pragma ADDRESS trcmr_addr 0120H /* Timer RC mode register */
#pragma ADDRESS trccr1_addr 0121H /* Timer RC control register 1 */
#pragma ADDRESS trcier_addr 0122H /* Timer RC interrupt enable register */
#pragma ADDRESS trcsr_addr 0123H /* Timer RC status register */
#pragma ADDRESS trcior0_addr 0124H /* Timer RC I/O contorol register 0 */
#pragma ADDRESS trcior1_addr 0125H /* Timer RC I/O contorol register 1 */
#pragma ADDRESS trc_addr 0126H /* Timer RC counter register */
#pragma ADDRESS trcgra_addr 0128H /* Timer RC general register A */
#pragma ADDRESS trcgrb_addr 012AH /* Timer RC general register B */
#pragma ADDRESS trcgrc_addr 012CH /* Timer RC general register C */
#pragma ADDRESS trcgrd_addr 012EH /* Timer RC general register D */
#pragma ADDRESS trccr2_addr 0130H /* Timer RC control register 2 */
#pragma ADDRESS trcdf_addr 0131H /* Timer RC digital filter function selection
register */
#pragma ADDRESS trcoer_addr 0132H /* Timer RC output master enable register */
#pragma ADDRESS trcadcr_addr 0133H /* Timer RC trigger control register *//* 3x */
#pragma ADDRESS trdecr_addr 0135H /* Timer RD extended control register *//* 3x
*/
#pragma ADDRESS trdadcr_addr 0136H /* Timer RD trigger control register *//* 3x */
#pragma ADDRESS trdstr_addr 0137H /* Timer RD start register */
#pragma ADDRESS trdmr_addr 0138H /* Timer RD mode register */
#pragma ADDRESS trdpmr_addr 0139H /* Timer RD PWM mode register */
#pragma ADDRESS trdfcr_addr 013AH /* Timer RD function control register */
#pragma ADDRESS trdoer1_addr 013BH /* Timer RD output master enable register 1 */
#pragma ADDRESS trdoer2_addr 013CH /* Timer RD output master enable register 2 */
#pragma ADDRESS trdocr_addr 013DH /* Timer RD output control register */
#pragma ADDRESS trddf0_addr 013EH /* Timer RD digital filter function contorol
```

```
register 0 */
#pragma ADDRESS trddf1_addr 013FH /* Timer RD digital filter function contorol register 1 */
#pragma ADDRESS trdcr0_addr 0140H /* Timer RD control register 0 */
#pragma ADDRESS trdiora0_addr 0141H /* Timer RD I/O contorol register A0 */
#pragma ADDRESS trdiorc0_addr 0142H /* Timer RD I/O contorol register C0 */
#pragma ADDRESS trdsr0_addr 0143H /* Timer RD status register 0 */
#pragma ADDRESS trdier0_addr 0144H /* Timer RD interrupt enable register 0 */
#pragma ADDRESS trdpocr0_addr 0145H /* Timer RD PWM mode output level control register 0 */
#pragma ADDRESS trd0_addr 0146H /* Timer RD counter 0 */
#pragma ADDRESS trdgra0_addr 0148H /* Timer RD general register A0 */
#pragma ADDRESS trdgrb0_addr 014AH /* Timer RD general register B0 */
#pragma ADDRESS trdgrc0_addr 014CH /* Timer RD general register C0 */
#pragma ADDRESS trdgrd0_addr 014EH /* Timer RD general register D0 */
#pragma ADDRESS trdcr1_addr 0150H /* Timer RD control register 1 */
#pragma ADDRESS trdiora1_addr 0151H /* Timer RD I/O contorol register A1 */
#pragma ADDRESS trdiorc1_addr 0152H /* Timer RD I/O contorol register C1 */
#pragma ADDRESS trdsr1_addr 0153H /* Timer RD status register 1 */
#pragma ADDRESS trdier1_addr 0154H /* Timer RD interrupt enable register 1 */
#pragma ADDRESS trdpocr1_addr 0155H /* Timer RD PWM mode output level control register 1 */
#pragma ADDRESS trd1_addr 0156H /* Timer RD counter 1 */
#pragma ADDRESS trdgra1_addr 0158H /* Timer RD general register A1 */
#pragma ADDRESS trdgrb1_addr 015AH /* Timer RD general register B1 */
#pragma ADDRESS trdgrc1_addr 015CH /* Timer RD general register C1 */
#pragma ADDRESS trdgrd1_addr 015EH /* Timer RD general register D1 */
#pragma ADDRESS u1mr_addr 0160H /* UART1 transmit/receive mode register *//* 3x */
#pragma ADDRESS u1brg_addr 0161H /* UART1 bit rate register *//* 3x */
#pragma ADDRESS u1tb_addr 0162H /* UART1 transmit buffer register *//* 3x */
#pragma ADDRESS u1c0_addr 0164H /* UART1 transmit/receive control register 0 *//* 3x */
#pragma ADDRESS u1c1_addr 0165H /* UART1 transmit/receive control register 1 *//* 3x */
#pragma ADDRESS u1rb_addr 0166H /* UART1 receive buffer register *//* 3x */
#pragma ADDRESS trasr_addr 0180H /* Timer RA function select register *//* 3x */
#pragma ADDRESS trbrcsr_addr 0181H /* Timer RB/RC function select register *//* 3x */
#pragma ADDRESS trcpsr0_addr 0182H /* Timer RC function select register 0 *//* 3x */
#pragma ADDRESS trcpsr1_addr 0183H /* Timer RC function select register 1 *//* 3x */
#pragma ADDRESS trdpsr0_addr 0184H /* Timer RD function select register 0 *//* 3x */
#pragma ADDRESS trdpsr1_addr 0185H /* Timer RD function select register 1 *//* 3x */
#pragma ADDRESS timsr_addr 0186H /* Timer pin select register *//* 3x */
#pragma ADDRESS u0sr_addr 0188H /* UART0 function select register *//* 3x */
#pragma ADDRESS u1sr_addr 0189H /* UART1 function select register *//* 3x */
#pragma ADDRESS u2sr0_addr 018AH /* UART2 function select register 0 *//* 3x */
#pragma ADDRESS u2sr1_addr 018BH /* UART2 function select register 1 *//* 3x */
#pragma ADDRESS ssuiicsr_addr 018CH /* SSU/IIC function select register *//* 3x */
#pragma ADDRESS intsr_addr 018EH /* INT function select register *//* 3x */
#pragma ADDRESS pinsr_addr 018FH /* INPUT/OUTPUT function select register *//* 3x */
#pragma ADDRESS ssbr_addr 0193H /* SS bit counter register *//* 3x */
#pragma ADDRESS sstdr_addr 0194H /* SS transmit data register Low *//* 3x */
#pragma ADDRESS icdrt_addr 0194H /* IIC bus transmit data register *//* 3x */
#pragma ADDRESS sstdrh_addr 0195H /* SS transmit data register High *//* 3x */
#pragma ADDRESS ssrdr_addr 0196H /* SS receive data register Low *//* 3x */
#pragma ADDRESS icdrr_addr 0196H /* IIC bus receive data register *//* 3x */
#pragma ADDRESS ssrdrh_addr 0197H /* SS receive data register High *//* 3x */
#pragma ADDRESS sscrh_addr 0198H /* SS control register H *//* 3x */
#pragma ADDRESS iccr1_addr 0198H /* IIC bus control register 1 *//* 3x */
#pragma ADDRESS sscrl_addr 0199H /* SS control register L *//* 3x */
#pragma ADDRESS iccr2_addr 0199H /* IIC bus control register 2 *//* 3x */
#pragma ADDRESS ssmr_addr 019AH /* SS mode register *//* 3x */
#pragma ADDRESS icmr_addr 019AH /* IIC bus mode register *//* 3x */
#pragma ADDRESS sser_addr 019BH /* SS enable register *//* 3x */
#pragma ADDRESS icier_addr 019BH /* IIC bus interrupt enable register *//* 3x */
```

```
#pragma ADDRESS		sssr_addr		019CH	/* SS status register *//* 3x */
#pragma ADDRESS		icsr_addr		019CH	/* IIC bus status register *//* 3x */
#pragma ADDRESS		ssmr2_addr			019DH	/* SS mode register 2 *//* 3x */
#pragma ADDRESS		sar_addr		019DH	/* Slave ddress register *//* 3x */
#pragma ADDRESS		fst_addr		01B2H	/* Flash memory status register *//* 3x */
#pragma ADDRESS		fmr0_addr			01B4H	/* Flash memory control register 0 *//* 3x */
#pragma ADDRESS		fmr1_addr			01B5H	/* Flash memory control register 1 */
#pragma ADDRESS		fmr2_addr			01B6H	/* Flash memory control register 2 *//* 3x */
#pragma ADDRESS		rmad0_addr			01C0H	/* Address match interrupt register0 */
#pragma ADDRESS		aier0_addr			01C3H	/* Address match interrupt enable register 0
*/
#pragma ADDRESS		rmad1_addr			01C4H	/* Address match interrupt register1 */
#pragma ADDRESS		aier1_addr			01C7H	/* Address match interrupt enable register 1
*//* 3x */
#pragma ADDRESS		pur0_addr			01E0H	/* Pull-up control register 0 *//* 3x */
#pragma ADDRESS		pur1_addr			01E1H	/* Pull-up control register 1 *//* 3x */
#pragma ADDRESS		p1drr_addr			01F0H	/* Port P1 drivability control register *//* 3x
*/
#pragma ADDRESS		p2drr_addr			01F1H	/* Port P2 drivability control register *//* 3x
*/
#pragma ADDRESS		drr0_addr		01F2H	/* Drivability control register 0 *//* 3x */
#pragma ADDRESS		drr1_addr		01F3H	/* Drivability control register 1 *//* 3x */
#pragma ADDRESS		vlt0_addr		01F5H	/* Input threshold control register 0 *//* 3x */
#pragma ADDRESS		vlt1_addr		01F6H	/* Input threshold control register 1 *//* 3x */
#pragma ADDRESS		intcmp_addr			01F8H	/* comparator B control register *//* 3x */
#pragma ADDRESS		inten_addr			01FAH	/* external input enable register 0 *//* 3x */
#pragma ADDRESS		inten1_addr			01FBH	/* external input enable register 1    *//* 3x */
#pragma ADDRESS		intf_addr		01FCH	/* INT input filter select register 0 *//* 3x */
#pragma ADDRESS		intf1_addr			01FDH	/* INT input filter select register 1 *//* 3x */
#pragma ADDRESS		kien_addr			01FEH	/* Key input enable register 0 *//* 3x */
#pragma ADDRESS		dtcd0_addr			2C40H	/* DTC control data 0 *//* 3x */
#pragma ADDRESS		dtcd1_addr			2C48H	/* DTC control data 1 *//* 3x */
#pragma ADDRESS		dtcd2_addr			2C50H	/* DTC control data 2 *//* 3x */
#pragma ADDRESS		dtcd3_addr			2C58H	/* DTC control data 3 *//* 3x */
#pragma ADDRESS		dtcd4_addr			2C60H	/* DTC control data 4 *//* 3x */
#pragma ADDRESS		dtcd5_addr			2C68H	/* DTC control data 5 *//* 3x */
#pragma ADDRESS		dtcd6_addr			2C70H	/* DTC control data 6 *//* 3x */
#pragma ADDRESS		dtcd7_addr			2C78H	/* DTC control data 7 *//* 3x */
#pragma ADDRESS		dtcd8_addr			2C80H	/* DTC control data 8 *//* 3x */
#pragma ADDRESS		dtcd9_addr			2C88H	/* DTC control data 9 *//* 3x */
#pragma ADDRESS		dtcd10_addr			2C90H	/* DTC control data 10 *//* 3x */
#pragma ADDRESS		dtcd11_addr			2C98H	/* DTC control data 11 *//* 3x */
#pragma ADDRESS		dtcd12_addr			2CA0H	/* DTC control data 12 *//* 3x */
#pragma ADDRESS		dtcd13_addr			2CA8H	/* DTC control data 13 *//* 3x */
#pragma ADDRESS		dtcd14_addr			2CB0H	/* DTC control data 14 *//* 3x */
#pragma ADDRESS		dtcd15_addr			2CB8H	/* DTC control data 15 *//* 3x */
#pragma ADDRESS		dtcd16_addr			2CC0H	/* DTC control data 16 *//* 3x */
#pragma ADDRESS		dtcd17_addr			2CC8H	/* DTC control data 17 *//* 3x */
#pragma ADDRESS		dtcd18_addr			2CD0H	/* DTC control data 18 *//* 3x */
#pragma ADDRESS		dtcd19_addr			2CD8H	/* DTC control data 19 *//* 3x */
#pragma ADDRESS		dtcd20_addr			2CE0H	/* DTC control data 20 *//* 3x */
#pragma ADDRESS		dtcd21_addr			2CE8H	/* DTC control data 21 *//* 3x */
#pragma ADDRESS		dtcd22_addr			2CF0H	/* DTC control data 22 *//* 3x */
#pragma ADDRESS		dtcd23_addr			2CF8H	/* DTC control data 23 *//* 3x */

/************************************************************
*  declare SFR bit                                          *
************************************************************/
struct	bit_def {
		char	b0:1;
		char	b1:1;
		char	b2:1;
		char	b3:1;
		char	b4:1;
		char	b5:1;
		char	b6:1;
		char	b7:1;
};
union	byte_def{
		struct	bit_def bit;
		char	byte;
};

/*-----------------------------------------------
```

```
   Processor mode register0
-----------------------------------------------------*/
union     byte_def  pm0_addr;
#define             pm0                   pm0_addr.byte

#define             pm03                  pm0_addr.bit.b3          /* Software reset bit */

/*-----------------------------------------------------
   Processor mode register1
-----------------------------------------------------*/
union     byte_def  pm1_addr;
#define             pm1                   pm1_addr.byte

#define             pm12                  pm1_addr.bit.b2          /* WDT interrupt/reset select bit */

/*-----------------------------------------------------
   System clock control register0
-----------------------------------------------------*/
union     byte_def  cm0_addr;
#define             cm0                   cm0_addr.byte

#define             cm01                  cm0_addr.bit.b1          /* Xin-Xcin switch bits */
#define             cm02                  cm0_addr.bit.b2          /* WAIT peripheral function clock stop bit */
#define             cm03                  cm0_addr.bit.b3          /* Main clock (Xcin) stop bit */
#define             cm04                  cm0_addr.bit.b4          /* Port Xcin-Xcout Switch bit */
#define             cm05                  cm0_addr.bit.b5          /* Main clock (Xin-Xout) stop bit */
#define             cm06                  cm0_addr.bit.b6          /* Main clock division select bit0 */
#define             cm07                  cm0_addr.bit.b7          /* CPU Clock Select bit */

/*-----------------------------------------------------
   System clock control register1
-----------------------------------------------------*/
union     byte_def  cm1_addr;
#define             cm1                   cm1_addr.byte

#define             cm10                  cm1_addr.bit.b0          /* All clock stop control bit */
#define             cm11                  cm1_addr.bit.b1          /* XIN-XOUT internal resistor select bit */
#define             cm12                  cm1_addr.bit.b2          /* XCIN-XCOUT internal resistor select bit
*/
#define             cm13                  cm1_addr.bit.b3          /* Port Xin-Xout switch bits */
#define             cm14                  cm1_addr.bit.b4          /* Low-speed on-chip oscillator stop bit */
#define             cm16                  cm1_addr.bit.b6          /* Main clock division select bit1 */
#define             cm17                  cm1_addr.bit.b7          /* Main clock division select bit1 */

/*-----------------------------------------------------
   Module clock enable register
-----------------------------------------------------*/
union     byte_def  mstcr_addr;
#define             mstcr                 mstcr_addr.byte

#define             mstiic                mstcr_addr.bit.b3        /* SSU/IIC operation enable bit */
#define             msttrd                mstcr_addr.bit.b4        /* Timer RD operation enable bit */
#define             msttrc                mstcr_addr.bit.b5        /* Timer RC operation enable bit */

/*-----------------------------------------------------
   System clock control register 3
-----------------------------------------------------*/
union     byte_def  cm3_addr;
#define             cm3                   cm3_addr.byte

#define             cm30                  cm3_addr.bit.b0          /* WAIT control bit */
#define             cm35                  cm3_addr.bit.b5          /*  CPU  clock  division  select  bit  for
WAIT/STOP return */
#define             cm36                  cm3_addr.bit.b6          /*  CPU  Clock  Select  bit  for  WAIT/STOP
return */
#define             cm37                  cm3_addr.bit.b7          /*  CPU  Clock  Select  bit  for  WAIT/STOP
return */

/*-----------------------------------------------------
   High-speed on-chip oscillator control register 7
-----------------------------------------------------*/
union     byte_def  fra7_addr;
#define             fra7                  fra7_addr.byte

/*-----------------------------------------------------
   Protect register
-----------------------------------------------------*/
union     byte_def  prcr_addr;
#define             prcr                  prcr_addr.byte

#define             prc0                  prcr_addr.bit.b0         /* Protect bit0 */
#define             prc1                  prcr_addr.bit.b1         /* Protect bit1 */
#define             prc2                  prcr_addr.bit.b2         /* Protect bit2 */
#define             prc3                  prcr_addr.bit.b3         /* Protect bit3 */

/*-----------------------------------------------------
   Reset factor register
-----------------------------------------------------*/
union     byte_def  rstfr_addr;
#define             rstfr                 rstfr_addr.byte

#define             cwr                       rstfr_addr.bit.b0    /* Cold/Warm Start reception flag */
#define             hwr                       rstfr_addr.bit.b1    /* H/W Reset detection flag */
#define             swr                       rstfr_addr.bit.b2    /* S/W Reset detection flag */
#define             wdr                       rstfr_addr.bit.b3    /* WDT Reset detection flag */

/*-----------------------------------------------------
   Oscillation stop detection register
-----------------------------------------------------*/
union     byte_def  ocd_addr;
#define             ocd                       ocd_addr.byte

#define             ocd0                  ocd_addr.bit.b0          /* Oscillation stop detection enable bit */
#define             ocd1                  ocd_addr.bit.b1          /* Oscillation stop detection interrupt enable
bit */
#define             ocd2                  ocd_addr.bit.b2          /* System clock select bit */
#define             ocd3                  ocd_addr.bit.b3          /* Clock monitor bit */

/*-----------------------------------------------------
   Watchdog timer reset register
-----------------------------------------------------*/
union     byte_def  wdtr_addr;
#define             wdtr                  wdtr_addr.byte

/*-----------------------------------------------------
   Watchdog timer start register
-----------------------------------------------------*/
union     byte_def  wdts_addr;
#define             wdts                  wdts_addr.byte

/*-----------------------------------------------------
   Watchdog timer control register
-----------------------------------------------------*/
union     byte_def  wdtc_addr;
```

```
#define             wdtc                wdtc_addr.byte

#define             wdtc7               wdtc_addr.bit.b7                /* Prescaler select bit */

/*-----------------------------------------------------
   Count source protect mode register
-----------------------------------------------------*/
union     byte_def  cspr_addr;
#define             cspr                cspr_addr.byte
#define             cspro               cspr_addr.bit.b7            /* WDT count source protect mode select bit */

/*-----------------------------------------------------
   High-speed on-chip oscillator control register 0
-----------------------------------------------------*/
union     byte_def  fra0_addr;
#define             fra0                fra0_addr.byte

#define             fra00               fra0_addr.bit.b0            /* High-speed on-chip oscillator enable bit */
#define             fra01               fra0_addr.bit.b1            /* High-speed on-chip oscillator select bit */

/*-----------------------------------------------------
   High-speed on-chip oscillator control register 1
-----------------------------------------------------*/
union     byte_def  fra1_addr;
#define             fra1                fra1_addr.byte

/*-----------------------------------------------------
   High-speed on-chip oscillator control register 2
-----------------------------------------------------*/
union     byte_def  fra2_addr;
#define             fra2                fra2_addr.byte

#define             fra20               fra2_addr.bit.b0            /* High-speed on-chip oscillator frequency switching bit
*/
#define             fra21               fra2_addr.bit.b1            /* High-speed on-chip oscillator frequency switching bit
*/
#define             fra22               fra2_addr.bit.b2            /* High-speed on-chip oscillator frequency switching bit
*/

/*-----------------------------------------------------
   Internal voltage control register
-----------------------------------------------------*/
union     byte_def  ocvrefcr_addr;
#define             ocvrefan   ocvrefcr_addr.bit.b0

/*-----------------------------------------------------
   Clock prescaler reset flag
-----------------------------------------------------*/
union     byte_def  cpsrf_addr;
#define             cpsrf               cpsrf_addr.byte

#define             cpsr                cpsrf_addr.bit.b7           /* Clock prescaler reset flag */

/*-----------------------------------------------------
   High-speed on-chip oscillator control register 4
-----------------------------------------------------*/
union     byte_def  fra4_addr;
#define             fra4                fra4_addr.byte

/*-----------------------------------------------------
   High-speed on-chip oscillator control register 5
-----------------------------------------------------*/
union     byte_def  fra5_addr;
#define             fra5                fra5_addr.byte

/*-----------------------------------------------------
   High-speed on-chip oscillator control register 6
-----------------------------------------------------*/
union     byte_def  fra6_addr;
#define             fra6                fra6_addr.byte

/*-----------------------------------------------------
   High-speed on-chip oscillator control register 3
-----------------------------------------------------*/
union     byte_def  fra3_addr;
#define             fra3                fra3_addr.byte

/*-----------------------------------------------------
   Voltage monitor circuit / comparator A control register
-----------------------------------------------------*/
union     byte_def  cmpa_addr;
#define             cmpa                cmpa_addr.byte

#define             cm1por              cmpa_addr.bit.b0            /* LVCOUT1 output polarity select bit */
#define             cm2por              cmpa_addr.bit.b1            /* LVCOUT2 output polarity select bit */
#define             cm1oe               cmpa_addr.bit.b2            /* LVCOUT1 output enable bit */
#define             cm2oe               cmpa_addr.bit.b3            /* LVCOUT2 output enable bit */
#define             irq1sel             cmpa_addr.bit.b4            /* Voltage monitor circuit1 / comparator A1 interrupt
select bit */
#define             irq2sel             cmpa_addr.bit.b5            /* Voltage monitor circuit2 / comparator A2 interrupt
select bit */
#define             compsel             cmpa_addr.bit.b7            /* irq1sel/irq2sel enable bit */

/*-----------------------------------------------------
   Voltage monitor circuit edge select register
-----------------------------------------------------*/
union     byte_def  vcac_addr;
#define             vcac                vcac_addr.byte

#define             vcac1               vcac_addr.bit.b1            /* comparator A1 edge selection bit */
#define             vcac2               vcac_addr.bit.b2            /* comparator A2 edge selection bit */

/*-----------------------------------------------------
   Voltage detection register 1
-----------------------------------------------------*/
union     byte_def  vca1_addr;
#define             vca1                vca1_addr.byte

#define             vca13               vca1_addr.bit.b3            /* Voltage detection 2 signal monitor flag */

/*-----------------------------------------------------
   Voltage detection register 2
-----------------------------------------------------*/
union     byte_def  vca2_addr;
#define             vca2                vca2_addr.byte

#define             vca20               vca2_addr.bit.b0            /* Internal power supply low power consumption
enable bit */
#define             vca21               vca2_addr.bit.b1            /* comparator A 1 input Voltage selection bit */
#define             vca22               vca2_addr.bit.b2            /* LVCMP1 external input select bit */
#define             vca23               vca2_addr.bit.b3            /* comparator A 2 input Voltage selection bit */
#define             vca24               vca2_addr.bit.b4            /* LVCMP2 external input select bit */
#define             vca25               vca2_addr.bit.b5            /* Voltage detection 0 enable bit */
#define             vca26               vca2_addr.bit.b6            /* Voltage detection 1/ comparator A 1 enable bit */
```

```
#define         vca27           vca2_addr.bit.b7        /* Voltage detection 2/ comparator A 2 enable bit */

/*-----------------------------------------------------
   Voltage detection 1 level select register
-----------------------------------------------------*/
union   byte_def  vd1ls_addr;
#define         vd1ls           vd1ls_addr.byte

#define         vd1s0           vd1ls_addr.bit.b0       /* Voltage detection 1 Level select bit */
#define         vd1s1           vd1ls_addr.bit.b1       /* Voltage detection 1 Level select bit */
#define         vd1s2           vd1ls_addr.bit.b2       /* Voltage detection 1 Level select bit */
#define         vd1s3           vd1ls_addr.bit.b3       /* Voltage detection 1 Level select bit */

/*-----------------------------------------------------
   Voltage monitor 0 circuit control register
-----------------------------------------------------*/
union   byte_def  vw0c_addr;
#define         vw0c            vw0c_addr.byte

#define         vw0c0           vw0c_addr.bit.b0        /* Voltage monitor 0 reset enable bit */
#define         vw0c1           vw0c_addr.bit.b1        /* Voltage monitor 0 digital filter disabled mode select
bit */
#define         vw0f0           vw0c_addr.bit.b4        /* Sampling clock select bit */
#define         vw0f1           vw0c_addr.bit.b5        /* Sampling clock select bit */

/*-----------------------------------------------------
   Voltage monitor 1 circuit control register
-----------------------------------------------------*/
union   byte_def  vw1c_addr;
#define         vw1c            vw1c_addr.byte

#define         vw1c0           vw1c_addr.bit.b0        /* Voltage monitor 1 interrupt enable bit */
#define         vw1c1           vw1c_addr.bit.b1        /* Voltage Monitor 1 digital filter disable mode select
bit */
#define         vw1c2           vw1c_addr.bit.b2        /* Voltage change detection flag */
#define         vw1c3           vw1c_addr.bit.b3        /* Voltage detection 1 signal monitor flag */
#define         vw1f0           vw1c_addr.bit.b4        /* Sampling clock select bit */
#define         vw1f1           vw1c_addr.bit.b5        /* Sampling clock select bit */
#define         vw1c7           vw1c_addr.bit.b7        /* Voltage monitor 1 interrupt generating condition
select bit */

/*-----------------------------------------------------
   Voltage monitor 2 circuit control register
-----------------------------------------------------*/
union   byte_def  vw2c_addr;
#define         vw2c            vw2c_addr.byte

#define         vw2c0           vw2c_addr.bit.b0        /* Voltage monitor 2 interrupt enable bit */
#define         vw2c1           vw2c_addr.bit.b1        /* Voltage monitor 2 digital filter disabled mode select
bit */
#define         vw2c2           vw2c_addr.bit.b2        /* Voltage change detection flag */
#define         vw2c3           vw2c_addr.bit.b3        /* WDT Detection Flag */
#define         vw2f0           vw2c_addr.bit.b4        /* Sampling clock select bit */
#define         vw2f1           vw2c_addr.bit.b5        /* Sampling clock select bit */
#define         vw2c7           vw2c_addr.bit.b7        /* Voltage monitor 2 interrupt generating condition
select bit */

/*-----------------------------------------------------
   DTC start-up control register
-----------------------------------------------------*/
union   byte_def  dtctl_addr;
#define         dtctl           dtctl_addr.byte

#define         nmif            dtctl_addr.bit.b1       /* nm interrupt generating bit */

/*-----------------------------------------------------
   DTC start-up enable register 0
-----------------------------------------------------*/
union   byte_def  dtcen0_addr;
#define         dtcen0          dtcen0_addr.byte

#define         dtcen00         dtcen0_addr.bit.b0      /* DTC start-up enable bit */
#define         dtcen01         dtcen0_addr.bit.b1      /* DTC start-up enable bit */
#define         dtcen02         dtcen0_addr.bit.b2      /* DTC start-up enable bit */
#define         dtcen03         dtcen0_addr.bit.b3      /* DTC start-up enable bit */
#define         dtcen04         dtcen0_addr.bit.b4      /* DTC start-up enable bit */
#define         dtcen05         dtcen0_addr.bit.b5      /* DTC start-up enable bit */
#define         dtcen06         dtcen0_addr.bit.b6      /* DTC start-up enable bit */
#define         dtcen07         dtcen0_addr.bit.b7      /* DTC start-up enable bit */

/*-----------------------------------------------------
   DTC start-up enable register 1
-----------------------------------------------------*/
union   byte_def  dtcen1_addr;
#define         dtcen1          dtcen1_addr.byte

#define         dtcen10         dtcen1_addr.bit.b0      /* DTC start-up enable bit */
#define         dtcen11         dtcen1_addr.bit.b1      /* DTC start-up enable bit */
#define         dtcen12         dtcen1_addr.bit.b2      /* DTC start-up enable bit */
#define         dtcen13         dtcen1_addr.bit.b3      /* DTC start-up enable bit */
#define         dtcen14         dtcen1_addr.bit.b4      /* DTC start-up enable bit */
#define         dtcen15         dtcen1_addr.bit.b5      /* DTC start-up enable bit */
#define         dtcen16         dtcen1_addr.bit.b6      /* DTC start-up enable bit */
#define         dtcen17         dtcen1_addr.bit.b7      /* DTC start-up enable bit */

/*-----------------------------------------------------
   DTC start-up enable register 2
-----------------------------------------------------*/
union   byte_def  dtcen2_addr;
#define         dtcen2          dtcen2_addr.byte

#define         dtcen20         dtcen2_addr.bit.b0      /* DTC start-up enable bit */
#define         dtcen21         dtcen2_addr.bit.b1      /* DTC start-up enable bit */
#define         dtcen22         dtcen2_addr.bit.b2      /* DTC start-up enable bit */
#define         dtcen23         dtcen2_addr.bit.b3      /* DTC start-up enable bit */
#define         dtcen24         dtcen2_addr.bit.b4      /* DTC start-up enable bit */
#define         dtcen25         dtcen2_addr.bit.b5      /* DTC start-up enable bit */
#define         dtcen26         dtcen2_addr.bit.b6      /* DTC start-up enable bit */
#define         dtcen27         dtcen2_addr.bit.b7      /* DTC start-up enable bit */

/*-----------------------------------------------------
   DTC start-up enable register 3
-----------------------------------------------------*/
union   byte_def  dtcen3_addr;
#define         dtcen3          dtcen3_addr.byte

#define         dtcen30         dtcen3_addr.bit.b0      /* DTC start-up enable bit */
#define         dtcen31         dtcen3_addr.bit.b1      /* DTC start-up enable bit */
#define         dtcen32         dtcen3_addr.bit.b2      /* DTC start-up enable bit */
#define         dtcen33         dtcen3_addr.bit.b3      /* DTC start-up enable bit */
#define         dtcen34         dtcen3_addr.bit.b4      /* DTC start-up enable bit */
#define         dtcen35         dtcen3_addr.bit.b5      /* DTC start-up enable bit */
#define         dtcen36         dtcen3_addr.bit.b6      /* DTC start-up enable bit */
#define         dtcen37         dtcen3_addr.bit.b7      /* DTC start-up enable bit */
```

```
/*-----------------------------------------------------
   DTC start-up enable register 4
-----------------------------------------------------*/
union      byte_def  dtcen4_addr;
#define              dtcen4              dtcen4_addr.byte

#define              dtcen40             dtcen4_addr.bit.b0          /* DTC start-up enable bit */
#define              dtcen41             dtcen4_addr.bit.b1          /* DTC start-up enable bit */
#define              dtcen42             dtcen4_addr.bit.b2          /* DTC start-up enable bit */
#define              dtcen43             dtcen4_addr.bit.b3          /* DTC start-up enable bit */
#define              dtcen44             dtcen4_addr.bit.b4          /* DTC start-up enable bit */
#define              dtcen45             dtcen4_addr.bit.b5          /* DTC start-up enable bit */
#define              dtcen46             dtcen4_addr.bit.b6          /* DTC start-up enable bit */
#define              dtcen47             dtcen4_addr.bit.b7          /* DTC start-up enable bit */

/*-----------------------------------------------------
   DTC start-up enable register 5
-----------------------------------------------------*/
union      byte_def  dtcen5_addr;
#define              dtcen5              dtcen5_addr.byte

#define              dtcen50             dtcen5_addr.bit.b0          /* DTC start-up enable bit */
#define              dtcen51             dtcen5_addr.bit.b1          /* DTC start-up enable bit */
#define              dtcen52             dtcen5_addr.bit.b2          /* DTC start-up enable bit */
#define              dtcen53             dtcen5_addr.bit.b3          /* DTC start-up enable bit */
#define              dtcen54             dtcen5_addr.bit.b4          /* DTC start-up enable bit */
#define              dtcen55             dtcen5_addr.bit.b5          /* DTC start-up enable bit */
#define              dtcen56             dtcen5_addr.bit.b6          /* DTC start-up enable bit */
#define              dtcen57             dtcen5_addr.bit.b7          /* DTC start-up enable bit */

/*-----------------------------------------------------
   DTC start-up enable register 6
-----------------------------------------------------*/
union      byte_def  dtcen6_addr;
#define              dtcen6              dtcen6_addr.byte

#define              dtcen60             dtcen6_addr.bit.b0          /* DTC start-up enable bit */
#define              dtcen61             dtcen6_addr.bit.b1          /* DTC start-up enable bit */
#define              dtcen62             dtcen6_addr.bit.b2          /* DTC start-up enable bit */
#define              dtcen63             dtcen6_addr.bit.b3          /* DTC start-up enable bit */
#define              dtcen64             dtcen6_addr.bit.b4          /* DTC start-up enable bit */
#define              dtcen65             dtcen6_addr.bit.b5          /* DTC start-up enable bit */
#define              dtcen66             dtcen6_addr.bit.b6          /* DTC start-up enable bit */
#define              dtcen67             dtcen6_addr.bit.b7          /* DTC start-up enable bit */

/*-----------------------------------------------------
   UART0 bit rate register
-----------------------------------------------------*/
union      byte_def  u0brg_addr;
#define              u0brg               u0brg_addr.byte

/*-----------------------------------------------------
   UART2 bit rate register
-----------------------------------------------------*/
union      byte_def  u2brg_addr;
#define              u2brg               u2brg_addr.byte

/*-----------------------------------------------------
   UART2 degital filter function select register
-----------------------------------------------------*/
union      byte_def  urxdf_addr;
#define              urxdf               urxdf_addr.byte

#define              df2en               urxdf_addr.bit.b2           /* RXD2 digital filter enable bit */

/*-----------------------------------------------------
   UART2 special mode register 5
-----------------------------------------------------*/
union      byte_def  u2smr5_addr;
#define              u2smr5              u2smr5_addr.byte

#define              mp                  u2smr5_addr.bit.b0          /* Multiprocessor Communication enable bit
*/
#define              mpie                u2smr5_addr.bit.b4          /* Multiprocessor Communication control bit */

/*-----------------------------------------------------
   UART2 special mode register 4
-----------------------------------------------------*/
union      byte_def  u2smr4_addr;
#define              u2smr4              u2smr4_addr.byte

#define              stareq              u2smr4_addr.bit.b0          /* Start condition generation bit */
#define              rstareq             u2smr4_addr.bit.b1          /* Restart condition generation */
#define              stpreq              u2smr4_addr.bit.b2          /* Stop condition generation */
#define              stspsel             u2smr4_addr.bit.b3          /* SCL/SDA output select bit */
#define              ackd                u2smr4_addr.bit.b4          /* ACK data bit */
#define              ackc                u2smr4_addr.bit.b5          /* ACK data output enable bit */
#define              sclhi               u2smr4_addr.bit.b6          /* SCL stop enable bit */
#define              swc9                u2smr4_addr.bit.b7          /* SCL WAIT bit 3 */

/*-----------------------------------------------------
   UART2 special mode register 3
-----------------------------------------------------*/
union      byte_def  u2smr3_addr;
#define              u2smr3              u2smr3_addr.byte

#define              ckph                u2smr3_addr.bit.b1          /* clock phase set bit */
#define              nodc                u2smr3_addr.bit.b3          /* clock output select bit */
#define              dl0                 u2smr3_addr.bit.b5          /* DSA2 digital delay set bit */
#define              dl1                 u2smr3_addr.bit.b6          /* DSA2 digital delay set bit */
#define              dl2                 u2smr3_addr.bit.b7          /* DSA2 digital delay set bit */

/*-----------------------------------------------------
   UART2 special mode register 2
-----------------------------------------------------*/
union      byte_def  u2smr2_addr;
#define              u2smr2              u2smr2_addr.byte

#define              iicm2               u2smr2_addr.bit.b0          /* IIC mode select bit 2 */
#define              csc                 u2smr2_addr.bit.b1          /* clock synchronous bit */
#define              swc                 u2smr2_addr.bit.b2          /* SCL wait output bit */
#define              als                 u2smr2_addr.bit.b3          /* SDA output stop bit */
#define              stac                u2smr2_addr.bit.b4          /* UART2 INIT bit */
#define              swc2                u2smr2_addr.bit.b5          /* SCL wait output bit 2 */
#define              sdhi                u2smr2_addr.bit.b6          /* SDA output disabled bit */

/*-----------------------------------------------------
   UART2 special mode register
-----------------------------------------------------*/
union      byte_def  u2smr_addr;
#define              u2smr               u2smr_addr.byte

#define              iicm                u2smr_addr.bit.b0           /* IIC mode select bit */
```

```
#define         abc                 u2smr_addr.bit.b1       /* Arbitration lost detection flag control bit*/
#define         bbs                 u2smr_addr.bit.b2       /* Bus busy flag */
#define         abscs           u2smr_addr.bit.b4       /* Bus conflict detection sampling clock select bit */
#define         acse            u2smr_addr.bit.b5       /* Transmit enable bit auto clear select bit */
#define         sss                 u2smr_addr.bit.b6       /* Transmit satart condition select bit */

/*-----------------------------------------------------
  A-D mode register
-----------------------------------------------------*/
union   byte_def  admod_addr;
#define         admod           admod_addr.byte

#define         cks0            admod_addr.bit.b0       /* division select bit */
#define         cks1            admod_addr.bit.b1       /* division select bit */
#define         cks2            admod_addr.bit.b2       /* clock select bit */
#define         md0                 admod_addr.bit.b3       /* A-D mode select bit */
#define         md1                 admod_addr.bit.b4       /* A-D mode select bit */
#define         md2                 admod_addr.bit.b5       /* A-D mode select bit */
#define         adcap0          admod_addr.bit.b6       /* A-D Conversion trigger select bit */
#define         adcap1          admod_addr.bit.b7       /* A-D Conversion trigger select bit */

/*-----------------------------------------------------
  A-D input select register
-----------------------------------------------------*/
union   byte_def  adinsel_addr;
#define         adinsel         adinsel_addr.byte

#define         ch0                 adinsel_addr.bit.b0     /* Analog input pin select bit */
#define         ch1                 adinsel_addr.bit.b1     /* Analog input pin select bit */
#define         ch2                 adinsel_addr.bit.b2     /* Analog input pin select bit */
#define         scan0           adinsel_addr.bit.b4     /* A-D pin num select bit */
#define         scan1           adinsel_addr.bit.b5     /* A-D pin num select bit */
#define         adgsel0         adinsel_addr.bit.b6     /* A-D input group select bit */
#define         adgsel1         adinsel_addr.bit.b7     /* A-D input group select bit */

/*-----------------------------------------------------
  A-D control register0
-----------------------------------------------------*/
union   byte_def  adcon0_addr;
#define         adcon0          adcon0_addr.byte

#define         adst            adcon0_addr.bit.b0      /* A-D conversion start flag */

/*-----------------------------------------------------
  A-D control register1
-----------------------------------------------------*/
union   byte_def  adcon1_addr;
#define         adcon1          adcon1_addr.byte

#define         adex0           adcon1_addr.bit.b0      /* Expansion pin select bit */
#define         bits            adcon1_addr.bit.b4      /* 8/10-bit mode select bit */
#define         adstby          adcon1_addr.bit.b5      /* A-D standby bit */

/*-----------------------------------------------------
  D-A register 0
-----------------------------------------------------*/
union   byte_def  da0_addr;
#define         da0             da0_addr.byte

/*-----------------------------------------------------
  D-A register 1
-----------------------------------------------------*/
union   byte_def  da1_addr;
#define         da1             da1_addr.byte

/*-----------------------------------------------------
  D-A control register
-----------------------------------------------------*/
union   byte_def  dacon_addr;
#define         dacon           dacon_addr.byte

#define         da0e_dacon      dacon_addr.bit.b0       /* D/A 0 output enable bit */
#define         da1e_dacon      dacon_addr.bit.b1       /* D/A 1 output enable bit */

/*-----------------------------------------------------
  Port P0 register
-----------------------------------------------------*/
union   byte_def  p0_addr;
#define         p0              p0_addr.byte

#define         p0_0            p0_addr.bit.b0          /* Port P00 bit */
#define         p0_1            p0_addr.bit.b1          /* Port P01 bit */
#define         p0_2            p0_addr.bit.b2          /* Port P02 bit */
#define         p0_3            p0_addr.bit.b3          /* Port P03 bit */
#define         p0_4            p0_addr.bit.b4          /* Port P04 bit */
#define         p0_5            p0_addr.bit.b5          /* Port P05 bit */
#define         p0_6            p0_addr.bit.b6          /* Port P06 bit */
#define         p0_7            p0_addr.bit.b7          /* Port P07 bit */

/*-----------------------------------------------------
  Port P1 direction register
-----------------------------------------------------*/
union   byte_def  pd0_addr;
#define         pd0             pd0_addr.byte

#define         pd0_0           pd0_addr.bit.b0         /* Port P00 direction bit */
#define         pd0_1           pd0_addr.bit.b1         /* Port P01 direction bit */
#define         pd0_2           pd0_addr.bit.b2         /* Port P02 direction bit */
#define         pd0_3           pd0_addr.bit.b3         /* Port P03 direction bit */
#define         pd0_4           pd0_addr.bit.b4         /* Port P04 direction bit */
#define         pd0_5           pd0_addr.bit.b5         /* Port P05 direction bit */
#define         pd0_6           pd0_addr.bit.b6         /* Port P06 direction bit */
#define         pd0_7           pd0_addr.bit.b7         /* Port P07 direction bit */

/*-----------------------------------------------------
  Port P1 register
-----------------------------------------------------*/
union   byte_def  p1_addr;
#define         p1              p1_addr.byte

#define         p1_0            p1_addr.bit.b0          /* Port P10 bit */
#define         p1_1            p1_addr.bit.b1          /* Port P11 bit */
#define         p1_2            p1_addr.bit.b2          /* Port P12 bit */
#define         p1_3            p1_addr.bit.b3          /* Port P13 bit */
#define         p1_4            p1_addr.bit.b4          /* Port P14 bit */
#define         p1_5            p1_addr.bit.b5          /* Port P15 bit */
#define         p1_6            p1_addr.bit.b6          /* Port P16 bit */
#define         p1_7            p1_addr.bit.b7          /* Port P17 bit */

/*-----------------------------------------------------
  Port P1 direction register
-----------------------------------------------------*/
union   byte_def  pd1_addr;
#define         pd1             pd1_addr.byte
```

```
#define         pd1_0           pd1_addr.bit.b0          /* Port P10 direction bit */
#define         pd1_1           pd1_addr.bit.b1          /* Port P11 direction bit */
#define         pd1_2           pd1_addr.bit.b2          /* Port P12 direction bit */
#define         pd1_3           pd1_addr.bit.b3          /* Port P13 direction bit */
#define         pd1_4           pd1_addr.bit.b4          /* Port P14 direction bit */
#define         pd1_5           pd1_addr.bit.b5          /* Port P15 direction bit */
#define         pd1_6           pd1_addr.bit.b6          /* Port P16 direction bit */
#define         pd1_7           pd1_addr.bit.b7          /* Port P17 direction bit */

/*-----------------------------------------------------
   Port P2 register
-----------------------------------------------------*/
union     byte_def  p2_addr;
#define         p2                      p2_addr.byte

#define         p2_0            p2_addr.bit.b0           /* Port P20 bit */
#define         p2_1            p2_addr.bit.b1           /* Port P21 bit */
#define         p2_2            p2_addr.bit.b2           /* Port P22 bit */
#define         p2_3            p2_addr.bit.b3           /* Port P23 bit */
#define         p2_4            p2_addr.bit.b4           /* Port P24 bit */
#define         p2_5            p2_addr.bit.b5           /* Port P25 bit */
#define         p2_6            p2_addr.bit.b6           /* Port P26 bit */
#define         p2_7            p2_addr.bit.b7           /* Port P27 bit */

/*-----------------------------------------------------
   Port P2 direction register
-----------------------------------------------------*/
union     byte_def  pd2_addr;
#define         pd2                     pd2_addr.byte

#define         pd2_0           pd2_addr.bit.b0          /* Port P20 direction bit */
#define         pd2_1           pd2_addr.bit.b1          /* Port P21 direction bit */
#define         pd2_2           pd2_addr.bit.b2          /* Port P22 direction bit */
#define         pd2_3           pd2_addr.bit.b3          /* Port P23 direction bit */
#define         pd2_4           pd2_addr.bit.b4          /* Port P24 direction bit */
#define         pd2_5           pd2_addr.bit.b5          /* Port P25 direction bit */
#define         pd2_6           pd2_addr.bit.b6          /* Port P26 direction bit */
#define         pd2_7           pd2_addr.bit.b7          /* Port P27 direction bit */

/*-----------------------------------------------------
   Port P3 register
-----------------------------------------------------*/
union     byte_def  p3_addr;
#define         p3                      p3_addr.byte

#define         p3_0            p3_addr.bit.b0           /* Port P30 bit */
#define         p3_1            p3_addr.bit.b1           /* Port P31 bit */
#define         p3_2            p3_addr.bit.b2           /* Port P32 bit */
#define         p3_3            p3_addr.bit.b3           /* Port P33 bit */
#define         p3_4            p3_addr.bit.b4           /* Port P34 bit */
#define         p3_5            p3_addr.bit.b5           /* Port P35 bit */
#define         p3_6            p3_addr.bit.b6           /* Port P36 bit */
#define         p3_7            p3_addr.bit.b7           /* Port P37 bit */

/*-----------------------------------------------------
   Port P3 direction register
-----------------------------------------------------*/
union     byte_def  pd3_addr;
#define         pd3                     pd3_addr.byte

#define         pd3_0           pd3_addr.bit.b0          /* Port P30 direction bit */
#define         pd3_1           pd3_addr.bit.b1          /* Port P31 direction bit */
#define         pd3_2           pd3_addr.bit.b2          /* Port P32 direction bit */
#define         pd3_3           pd3_addr.bit.b3          /* Port P33 direction bit */
#define         pd3_4           pd3_addr.bit.b4          /* Port P34 direction bit */
#define         pd3_5           pd3_addr.bit.b5          /* Port P35 direction bit */
#define         pd3_6           pd3_addr.bit.b6          /* Port P36 direction bit */
#define         pd3_7           pd3_addr.bit.b7          /* Port P37 direction bit */

/*-----------------------------------------------------
   Port P4 register
-----------------------------------------------------*/
union     byte_def  p4_addr;
#define         p4                      p4_addr.byte

#define         p4_2            p4_addr.bit.b2           /* Port P42 bit */
#define         p4_3            p4_addr.bit.b3           /* Port P43 bit */
#define         p4_4            p4_addr.bit.b4           /* Port P44 bit */
#define         p4_5            p4_addr.bit.b5           /* Port P45 bit */
#define         p4_6            p4_addr.bit.b6           /* Port P46 bit */
#define         p4_7            p4_addr.bit.b7           /* Port P47 bit */

/*-----------------------------------------------------
   Port P5 register
-----------------------------------------------------*/
union     byte_def  p5_addr;
#define         p5                      p5_addr.byte

#define         p5_6            p5_addr.bit.b6           /* Port P56 bit */
#define         p5_7            p5_addr.bit.b7           /* Port P57 bit */

/*-----------------------------------------------------
   Port P4 direction register
-----------------------------------------------------*/
union     byte_def  pd4_addr;
#define         pd4                     pd4_addr.byte

#define         pd4_2           pd4_addr.bit.b2          /* Port P42 direction bit */
#define         pd4_3           pd4_addr.bit.b3          /* Port P43 direction bit */
#define         pd4_4           pd4_addr.bit.b4          /* Port P44 direction bit */
#define         pd4_5           pd4_addr.bit.b5          /* Port P45 direction bit */
#define         pd4_6           pd4_addr.bit.b6          /* Port P46 direction bit */
#define         pd4_7           pd4_addr.bit.b7          /* Port P47 direction bit */

/*-----------------------------------------------------
   Port P5 direction register
-----------------------------------------------------*/
union     byte_def  pd5_addr;
#define         pd5                     pd5_addr.byte

#define         pd5_6           pd5_addr.bit.b6          /* Port P56 direction bit */
#define         pd5_7           pd5_addr.bit.b7          /* Port P57 direction bit */

/*-----------------------------------------------------
   Port P6 register
-----------------------------------------------------*/
union     byte_def  p6_addr;
#define         p6                      p6_addr.byte

#define         p6_0            p6_addr.bit.b0           /* Port P60 bit */
#define         p6_1            p6_addr.bit.b1           /* Port P61 bit */
#define         p6_2            p6_addr.bit.b2           /* Port P62 bit */
#define         p6_3            p6_addr.bit.b3           /* Port P63 bit */
```

```
#define         p6_4            p6_addr.bit.b4                  /* Port P64 bit */
#define         p6_5            p6_addr.bit.b5                  /* Port P65 bit */
#define         p6_6            p6_addr.bit.b6                  /* Port P66 bit */
#define         p6_7            p6_addr.bit.b7                  /* Port P67 bit */

/*-----------------------------------------------------
  Port P6 direction register
-----------------------------------------------------*/
union   byte_def  pd6_addr;
#define         pd6                     pd6_addr.byte

#define         pd6_0           pd6_addr.bit.b0                 /* Port P60 direction bit */
#define         pd6_1           pd6_addr.bit.b1                 /* Port P61 direction bit */
#define         pd6_2           pd6_addr.bit.b2                 /* Port P62 direction bit */
#define         pd6_3           pd6_addr.bit.b3                 /* Port P63 direction bit */
#define         pd6_4           pd6_addr.bit.b4                 /* Port P64 direction bit */
#define         pd6_5           pd6_addr.bit.b5                 /* Port P65 direction bit */
#define         pd6_6           pd6_addr.bit.b6                 /* Port P66 direction bit */
#define         pd6_7           pd6_addr.bit.b7                 /* Port P67 direction bit */

/*-----------------------------------------------------
  Timer RA control register
-----------------------------------------------------*/
union   byte_def  tracr_addr;
#define         tracr           tracr_addr.byte

#define         tstart_tracr    tracr_addr.bit.b0   /* Timer RA count start bit */
#define         tcstf_tracr             tracr_addr.bit.b1   /* Timer RA count status flag */
#define         tstop_tracr             tracr_addr.bit.b2   /* Timer RA count forcible stop bit */
#define         tedgf_tracr             tracr_addr.bit.b4   /* Active edge reception flag */
#define         tundf_tracr             tracr_addr.bit.b5   /* Timer RA underflow flag */

/*-----------------------------------------------------
  Timer RA I/O control register
-----------------------------------------------------*/
union   byte_def  traioc_addr;
#define         traioc          traioc_addr.byte

#define         tedgsel_traioc  traioc_addr.bit.b0  /* TRAIO polarity switch bit */
#define         topcr_traioc    traioc_addr.bit.b1  /* TRAIO output control bit */
#define         toena_traioc    traioc_addr.bit.b2  /* TRAO output enable bit */
#define         tiosel_traioc   traioc_addr.bit.b3  /* INT1/TRAIO select bit */
#define         tipf0_traioc    traioc_addr.bit.b4  /* TRAIO input filter select bit */
#define         tipf1_traioc    traioc_addr.bit.b5  /* TRAIO input filter select bit */
#define         tiogt0_traioc   traioc_addr.bit.b6  /* TRAIO event input control bit */
#define         tiogt1_traioc   traioc_addr.bit.b7  /* TRAIO event input control bit */

/*-----------------------------------------------------
  Timer RA mode register
-----------------------------------------------------*/
union   byte_def  tramr_addr;
#define         tramr           tramr_addr.byte

#define         tmod0_tramr             tramr_addr.bit.b0   /* Timer RA operating mode select bit */
#define         tmod1_tramr             tramr_addr.bit.b1   /* Timer RA operating mode select bit */
#define         tmod2_tramr             tramr_addr.bit.b2   /* Timer RA operating mode select bit */
#define         tck0_tramr              tramr_addr.bit.b4   /* Timer RA count source select bit */
#define         tck1_tramr              tramr_addr.bit.b5   /* Timer RA count source select bit */
#define         tck2_tramr              tramr_addr.bit.b6   /* Timer RA count source select bit */
#define         tckcut_tramr    tramr_addr.bit.b7   /* Timer RA count source cutoff bit */

/*-----------------------------------------------------
  Timer RA prescaler register
-----------------------------------------------------*/
union   byte_def  trapre_addr;
#define         trapre          trapre_addr.byte

/*-----------------------------------------------------
  Timer RA register
-----------------------------------------------------*/
union   byte_def  tra_addr;
#define         tra                     tra_addr.byte

/*-----------------------------------------------------
  LIN control register 2
-----------------------------------------------------*/
union   byte_def  lincr2_addr;
#define         lincr2          lincr2_addr.byte

#define         bce_lincr2              lincr2_addr.bit.b0   /* Bus conflict detection select bit for Synch Break
Transmit */

/*-----------------------------------------------------
  LIN control register
-----------------------------------------------------*/
union   byte_def  lincr_addr;
#define         lincr           lincr_addr.byte

#define         sfie_lincr      lincr_addr.bit.b0   /* Synchronous field measurementcompleted interrupt enable bit
*/
#define         sbie_lincr      lincr_addr.bit.b1   /* Synchronous break detection interrupt enable bit */
#define         bcie_lincr      lincr_addr.bit.b2   /* Bus conflict detection interrupt enable bit */
#define         rxdsf_lincr             lincr_addr.bit.b3   /* RxD0 input status flag */
#define         lstart_lincr    lincr_addr.bit.b4   /* Synchronous Break detection start bit */
#define         sbe_lincr       lincr_addr.bit.b5   /* RxD0 input unmasking timing select bit */
#define         mst_lincr       lincr_addr.bit.b6   /* LIN operation mode setting bit */
#define         line_lincr      lincr_addr.bit.b7   /* LIN operation start bit */

/*-----------------------------------------------------
  LIN status register
-----------------------------------------------------*/
union   byte_def  linst_addr;
#define         linst           linst_addr.byte

#define         sfdct_linst             linst_addr.bit.b0   /* Synchronous field measurementcompleted flag */
#define         sbdct_linst             linst_addr.bit.b1   /* Synchronous break detection flag */
#define         bcdct_linst             linst_addr.bit.b2   /* Bus collision detection flag */
#define         b0clr_linst             linst_addr.bit.b3   /* SFDCT flag clear bit */
#define         b1clr_linst             linst_addr.bit.b4   /* SBDCT flag clear bit */
#define         b2clr_linst             linst_addr.bit.b5   /* BCDCT flag clear bit */

/*-----------------------------------------------------
  Timer RB control register
-----------------------------------------------------*/
union   byte_def  trbcr_addr;
#define         trbcr           trbcr_addr.byte

#define         tstart_trbcr    trbcr_addr.bit.b0   /* Timer RB count start bit */
#define         tcstf_trbcr             trbcr_addr.bit.b1   /* Timer RB count status flag */
#define         tstop_trbcr             trbcr_addr.bit.b2   /* Timer RB count forcible stop bit */

/*-----------------------------------------------------
  Timer RB one shot control register
-----------------------------------------------------*/
```

```
union      byte_def  trbocr_addr;
#define              trbocr              trbocr_addr.byte

#define              tosst_trbocr        trbocr_addr.bit.b0   /* Timer RB one-shot start bit */
#define              tossp_trbocr        trbocr_addr.bit.b1   /* Timer RB one-shot stop bit */
#define              tosstf_trbocr       trbocr_addr.bit.b2   /* Timer RB one-shot status flag */

/*-----------------------------------------------------
   Timer RB I/O control register
-----------------------------------------------------*/
union      byte_def  trbioc_addr;
#define              trbioc              trbioc_addr.byte

#define              topl_trbioc                 trbioc_addr.bit.b0   /* Timer RB output level select bit */
#define              tocnt_trbioc        trbioc_addr.bit.b1   /* Timer RB output switch bit */
#define              inostg_trbioc       trbioc_addr.bit.b2   /* One-shot trigger control bit */
#define              inoseg_trbioc       trbioc_addr.bit.b3   /* One-shot trigger polarity select bit */

/*-----------------------------------------------------
   Timer RB mode register
-----------------------------------------------------*/
union      byte_def  trbmr_addr;
#define              trbmr               trbmr_addr.byte

#define              tmod0_trbmr                 trbmr_addr.bit.b0   /* Timer RB operating mode select bit */
#define              tmod1_trbmr                 trbmr_addr.bit.b1   /* Timer RB operating mode select bit */
#define              twrc_trbmr                  trbmr_addr.bit.b3   /* Timer RB write control bit */
#define              tck0_trbmr                  trbmr_addr.bit.b4   /* Timer RB count source select bit */
#define              tck1_trbmr                  trbmr_addr.bit.b5   /* Timer RB count source select bit */
#define              tckcut_trbmr        trbmr_addr.bit.b7   /* Timer RB count source cutoff bit */

/*-----------------------------------------------------
   Timer RB prescaler register
-----------------------------------------------------*/
union      byte_def  trbpre_addr;
#define              trbpre              trbpre_addr.byte

/*-----------------------------------------------------
   Timer RB secondary register
-----------------------------------------------------*/
union      byte_def  trbsc_addr;
#define              trbsc               trbsc_addr.byte

/*-----------------------------------------------------
   Timer RB Primary Register
-----------------------------------------------------*/
union      byte_def  trbpr_addr;
#define              trbpr               trbpr_addr.byte

/*-----------------------------------------------------
   Timer RE seconds data register / Timer RE counter data register
-----------------------------------------------------*/
union      byte_def  tresec_addr;
#define              tresec              tresec_addr.byte

#define              sc00_tresec                 tresec_addr.bit.b0   /* 1st digit of seconds count bits */
#define              sc01_tresec                 tresec_addr.bit.b1   /* 1st digit of seconds count bits */
#define              sc02_tresec                 tresec_addr.bit.b2   /* 1st digit of seconds count bits */
#define              sc03_tresec                 tresec_addr.bit.b3   /* 1st digit of seconds count bits */
#define              sc10_tresec                 tresec_addr.bit.b4   /* 2nd digit of seconds count bits */
#define              sc11_tresec                 tresec_addr.bit.b5   /* 2nd digit of seconds count bits */
#define              sc12_tresec                 tresec_addr.bit.b6   /* 2nd digit of seconds count bits */
#define              bsy_tresec                  tresec_addr.bit.b7   /* Timer RE busy flag */

/*-----------------------------------------------------
   Timer RE minutes data register / Timer RE compare data register
-----------------------------------------------------*/
union      byte_def  tremin_addr;
#define              tremin              tremin_addr.byte

#define              mn00_tremin                 tremin_addr.bit.b0  /* 1st digit of minutes count bits */
#define              mn01_tremin                 tremin_addr.bit.b1  /* 1st digit of minutes count bits */
#define              mn02_tremin                 tremin_addr.bit.b2  /* 1st digit of minutes count bits */
#define              mn03_tremin                 tremin_addr.bit.b3  /* 1st digit of minutes count bits */
#define              mn10_tremin                 tremin_addr.bit.b4  /* 2nd digit of minutes count bits */
#define              mn11_tremin                 tremin_addr.bit.b5  /* 2nd digit of minutes count bits */
#define              mn12_tremin                 tremin_addr.bit.b6  /* 2nd digit of minutes count bits */
#define              bsy_tremin                  tremin_addr.bit.b7  /* Timer RE busy flag */

/*-----------------------------------------------------
   Timer RE Hours Data Register
-----------------------------------------------------*/
union      byte_def  trehr_addr;
#define              trehr               trehr_addr.byte

#define              hr00_trehr                  trehr_addr.bit.b0    /* 1st digit of hours count bits */
#define              hr01_trehr                  trehr_addr.bit.b1    /* 1st digit of hours count bits */
#define              hr02_trehr                  trehr_addr.bit.b2    /* 1st digit of hours count bits */
#define              hr03_trehr                  trehr_addr.bit.b3    /* 1st digit of hours count bits */
#define              hr10_trehr                  trehr_addr.bit.b4    /* 2nd digit of hours count bits */
#define              hr11_trehr                  trehr_addr.bit.b5    /* 2nd digit of hours count bits */
#define              bsy_trehr           trehr_addr.bit.b7    /* Timer RE busy flag */

/*-----------------------------------------------------
   Timer RE Days of Week Data Register
-----------------------------------------------------*/
union      byte_def  trewk_addr;
#define              trewk               trewk_addr.byte

#define              wk0_trewk                   trewk_addr.bit.b0   /* Days of week count bits */
#define              wk1_trewk                   trewk_addr.bit.b1   /* Days of week count bits */
#define              wk2_trewk                   trewk_addr.bit.b2   /* Days of week count bits */
#define              bsy_trewk                   trewk_addr.bit.b7   /* Timer RE busy flag */

/*-----------------------------------------------------
   Timer RE control register1
-----------------------------------------------------*/
union      byte_def  trecr1_addr;
#define              trecr1              trecr1_addr.byte

#define              tcstf_trecr1        trecr1_addr.bit.b1   /* Timer RE count status flag */
#define              toena_trecr1        trecr1_addr.bit.b2   /* TREO pin output enable bit */
#define              int_trecr1          trecr1_addr.bit.b3   /* Interrupt request timing bit */
#define              trerst_trecr1       trecr1_addr.bit.b4   /* Timer RE reset bit */
#define              pm_trecr1                   trecr1_addr.bit.b5   /* A.M. / P.M. bit */
#define              h12_h24_trecr1      trecr1_addr.bit.b6   /* Operating mode select bit */
#define              tstart_trecr1       trecr1_addr.bit.b7   /* Timer RE count start bit */

/*-----------------------------------------------------
   Timer RE control register2
-----------------------------------------------------*/
union      byte_def  trecr2_addr;
#define              trecr2              trecr2_addr.byte
```

```
#define             seie_trecr2                 trecr2_addr.bit.b0   /* Periodic interrupt triggered every second enable bit
*/
#define             mnie_trecr2                 trecr2_addr.bit.b1   /* Periodic interrupt triggered every minute enable bit
*/
#define             hrie_trecr2                 trecr2_addr.bit.b2   /* Periodic interrupt triggered every hour enable bit */
#define             dyie_trecr2                 trecr2_addr.bit.b3   /* Periodic interrupt triggered every day enable bit */
#define             wkie_trecr2                 trecr2_addr.bit.b4   /* Periodic interrupt triggered every week enable bit */
#define             comie_trecr2        trecr2_addr.bit.b5   /* Compare match interrupt enable bit */

/*-----------------------------------------------------
  Timer RE count source select register
-----------------------------------------------------*/
union     byte_def  trecsr_addr;
#define             trecsr              trecsr_addr.byte

#define             rcs0_trecsr                 trecsr_addr.bit.b0   /* Count source select bit */
#define             rcs1_trecsr                 trecsr_addr.bit.b1   /* Count source select bit */
#define             rcs2_trecsr                 trecsr_addr.bit.b2   /* 4-Bit counter select bit */
#define             rcs3_trecsr                 trecsr_addr.bit.b3   /* Real-Time clock mode select bit */
#define             rcs4_trecsr                 trecsr_addr.bit.b4   /* Clock output select bit */
#define             rcs5_trecsr                 trecsr_addr.bit.b5   /* Clock output select bit */
#define             rcs6_trecsr                 trecsr_addr.bit.b6   /* Clock output select bit */

/*-----------------------------------------------------
  Timer RC mode register
-----------------------------------------------------*/
union     byte_def  trcmr_addr;
#define             trcmr               trcmr_addr.byte

#define             pwmb_trcmr                  trcmr_addr.bit.b0    /* TRCIOB PWM mode select bit */
#define             pwmc_trcmr                  trcmr_addr.bit.b1    /* TRCIOC PWM mode select bit */
#define             pwmd_trcmr                  trcmr_addr.bit.b2    /* TRCIOD PWM mode select bit */
#define             pwm2_trcmr                  trcmr_addr.bit.b3    /* PWM2 mode select bit */
#define             bfc_trcmr           trcmr_addr.bit.b4    /* TRCGRC register function selection bit */
#define             bfd_trcmr           trcmr_addr.bit.b5    /* TRCGRD register function selection bit */
#define             tstart_trcmr        trcmr_addr.bit.b7    /* TRC count start bit */

/*-----------------------------------------------------
  Timer RC control register 1
-----------------------------------------------------*/
union     byte_def  trccr1_addr;
#define             trccr1              trccr1_addr.byte

#define             toa_trccr1          trccr1_addr.bit.b0   /* TRCIOA output level select bit */
#define             tob_trccr1          trccr1_addr.bit.b1   /* TRCIOB output level select bit */
#define             toc_trccr1          trccr1_addr.bit.b2   /* TRCIOC output level select bit */
#define             tod_trccr1                  trccr1_addr.bit.b3   /* TRCIOD output level select bit */
#define             tck0_trccr1                 trccr1_addr.bit.b4   /* Count source selection bit */
#define             tck1_trccr1                 trccr1_addr.bit.b5   /* Count source selection bit */
#define             tck2_trccr1                 trccr1_addr.bit.b6   /* Count source selection bit */
#define             cclr_trccr1                 trccr1_addr.bit.b7   /* TRC counter clear select bit */

/*-----------------------------------------------------
  Timer RC interrupt enable register
-----------------------------------------------------*/
union     byte_def  trcier_addr;
#define             trcier              trcier_addr.byte

#define             imiea_trcier        trcier_addr.bit.b0   /* Input capture / compare match interrupt enable bit A */
#define             imieb_trcier        trcier_addr.bit.b1   /* Input capture / compare match interrupt enable bit B */
#define             imiec_trcier        trcier_addr.bit.b2   /* Input capture / compare match interrupt enable bit C */
#define             imied_trcier        trcier_addr.bit.b3   /* Input capture / compare match interrupt enable bit D */
#define             ovie_trcier                 trcier_addr.bit.b7   /* Overflow / underflow interrupt enable bit */

/*-----------------------------------------------------
  Timer RC status register
-----------------------------------------------------*/
union     byte_def  trcsr_addr;
#define             trcsr               trcsr_addr.byte

#define             imfa_trcsr                  trcsr_addr.bit.b0    /* Input capture / compare match flag A */
#define             imfb_trcsr                  trcsr_addr.bit.b1    /* Input capture / compare match flag B */
#define             imfc_trcsr          trcsr_addr.bit.b2    /* Input capture / compare match flag C */
#define             imfd_trcsr                  trcsr_addr.bit.b3    /* Input capture / compare match flag D */
#define             ovf_trcsr           trcsr_addr.bit.b7    /* Overflow flag */

/*-----------------------------------------------------
  Timer RC I/O contorol register 0
-----------------------------------------------------*/
union     byte_def  trcior0_addr;
#define             trcior0             trcior0_addr.byte

#define             ioa0_trcior0        trcior0_addr.bit.b0  /* TRCGRA control bit */
#define             ioa1_trcior0        trcior0_addr.bit.b1  /* TRCGRA control bit */
#define             ioa2_trcior0        trcior0_addr.bit.b2  /* TRCGRA mode selection bit */
#define             iob0_trcior0        trcior0_addr.bit.b4  /* TRCGRB control bit */
#define             iob1_trcior0        trcior0_addr.bit.b5  /* TRCGRB control bit */
#define             iob2_trcior0        trcior0_addr.bit.b6  /* TRCGRB mode selection bit */

/*-----------------------------------------------------
  Timer RC I/O contorol register 1
-----------------------------------------------------*/
union     byte_def  trcior1_addr;
#define             trcior1             trcior1_addr.byte

#define             ioc0_trcior1        trcior1_addr.bit.b0  /* TRCGRC control bit */
#define             ioc1_trcior1        trcior1_addr.bit.b1  /* TRCGRC control bit */
#define             ioc2_trcior1        trcior1_addr.bit.b2  /* TRCGRC mode selection bit */
#define             ioc3_trcior1        trcior1_addr.bit.b3  /* TRCGRC register function selection bit */
#define             iod0_trcior1        trcior1_addr.bit.b4  /* TRCGRD control bit */
#define             iod1_trcior1        trcior1_addr.bit.b5  /* TRCGRD control bit */
#define             iod2_trcior1        trcior1_addr.bit.b6  /* TRCGRD mode selection bit */
#define             iod3_trcior1        trcior1_addr.bit.b7  /* TRCGRD register function selection bit */

/*-----------------------------------------------------
  Timer RC control register 2
-----------------------------------------------------*/
union     byte_def  trccr2_addr;
#define             trccr2              trccr2_addr.byte

#define             polb_trccr2                 trccr2_addr.bit.b0   /* PWM mode output level control bit B */
#define             polc_trccr2                 trccr2_addr.bit.b1   /* PWM mode output level control bit C */
#define             pold_trccr2                 trccr2_addr.bit.b2   /* PWM mode output level control bit D */
#define             cstp_trccr2                 trccr2_addr.bit.b5   /* Timer RC operating mode select bit */
#define             tceg0_trccr2        trccr2_addr.bit.b6   /* TRCTRG input edge selection bit */
#define             tceg1_trccr2        trccr2_addr.bit.b7   /* TRCTRG input edge selection bit */

/*-----------------------------------------------------
  Timer RC digital filter function selection register
-----------------------------------------------------*/
union     byte_def  trcdf_addr;
#define             trcdf               trcdf_addr.byte

#define             dfa_trcdf           trcdf_addr.bit.b0    /* TRCIOA pin digital filter function selection bit */
```

```
#define         dfb_trcdf       trcdf_addr.bit.b1   /* TRCIOB pin digital filter function selection bit */
#define         dfc_trcdf       trcdf_addr.bit.b2   /* TRCIOC pin digital filter function selection bit */
#define         dfd_trcdf       trcdf_addr.bit.b3   /* TRCIOD pin digital filter function selection bit */
#define         dftrg_trcdf             trcdf_addr.bit.b4   /* TRCIOG pin digital filter function selection bit */
#define         dfck0_trcdf             trcdf_addr.bit.b6   /* Clock selection bit for digital filter function */
#define         dfck1_trcdf             trcdf_addr.bit.b7   /* Clock selection bit for digital filter function */

/*-----------------------------------------------------
   Timer RC output master enable register
-----------------------------------------------------*/
union   byte_def  trcoer_addr;
#define         trcoer          trcoer_addr.byte

#define         ea_trcoer       trcoer_addr.bit.b0  /* TRCIOA output disable bit */
#define         eb_trcoer       trcoer_addr.bit.b1  /* TRCIOB output disable bit */
#define         ec_trcoer       trcoer_addr.bit.b2  /* TRCIOC output disable bit */
#define         ed_trcoer       trcoer_addr.bit.b3  /* TRCIOD output disable bit */
#define         pto_trcoer              trcoer_addr.bit.b7  /* INT0 of pulse output forced cutoff signal input
enabled bit */

/*-----------------------------------------------------
   Timer RC trigger control register
-----------------------------------------------------*/
union   byte_def  trcadcr_addr;
#define         trcadcr         trcadcr_addr.byte

#define         adtrgae_trcadcr trcadcr_addr.bit.b0 /* A-D trigger A enable bit */
#define         adtrgbe_trcadcr trcadcr_addr.bit.b1 /* A-D trigger B enable bit */
#define         adtrgce_trcadcr trcadcr_addr.bit.b2 /* A-D trigger C enable bit */
#define         adtrgde_trcadcr trcadcr_addr.bit.b3 /* A-D trigger D enable bit */

/*-----------------------------------------------------
   Timer RD extended control register
-----------------------------------------------------*/
union   byte_def  trdecr_addr;
#define         trdecr          trdecr_addr.byte

#define         itclk0_trdecr   trdecr_addr.bit.b3  /* fC2 select bit for CH0 */
#define         itclk1_trdecr   trdecr_addr.bit.b7  /* fC2 select bit for CH1 */

/*-----------------------------------------------------
   Timer RD trigger control register
-----------------------------------------------------*/
union   byte_def  trdadcr_addr;
#define         trdadcr         trdadcr_addr.byte

#define         adtrgd1e_trdadcr trdadcr_addr.bit.b0 /* A-D trigger D1 enable bit */
#define         adtrgc1e_trdadcr trdadcr_addr.bit.b1 /* A-D trigger C1 enable bit */
#define         adtrgb1e_trdadcr trdadcr_addr.bit.b2 /* A-D trigger B1 enable bit */
#define         adtrga1e_trdadcr trdadcr_addr.bit.b3 /* A-D trigger A1 enable bit */
#define         adtrgd0e_trdadcr trdadcr_addr.bit.b4 /* A-D trigger D0 enable bit */
#define         adtrgc0e_trdadcr trdadcr_addr.bit.b5 /* A-D trigger C0 enable bit */
#define         adtrgb0e_trdadcr trdadcr_addr.bit.b6 /* A-D trigger B0 enable bit */
#define         adtrga0e_trdadcr trdadcr_addr.bit.b7 /* A-D trigger A0 enable bit */

/*-----------------------------------------------------
   Timer RD start register
-----------------------------------------------------*/
union   byte_def  trdstr_addr;
#define         trdstr          trdstr_addr.byte

#define         tstart0_trdstr  trdstr_addr.bit.b0  /* TRD0 count start bit */
#define         tstart1_trdstr  trdstr_addr.bit.b1  /* TRD1 count start bit */
#define         csel0_trdstr    trdstr_addr.bit.b2  /* TRD0 count operating mode select bit */
#define         csel1_trdstr    trdstr_addr.bit.b3  /* TRD1 count operating mode select bit */

/*-----------------------------------------------------
   Timer RD mode register
-----------------------------------------------------*/
union   byte_def  trdmr_addr;
#define         trdmr           trdmr_addr.byte

#define         sync_trdmr              trdmr_addr.bit.b0   /* Timer RD synchronous bit */
#define         bfc0_trdmr              trdmr_addr.bit.b4   /* GRC0 register function selection bit */
#define         bfd0_trdmr              trdmr_addr.bit.b5   /* GRD0 register function selection bit */
#define         bfc1_trdmr              trdmr_addr.bit.b6   /* GRC1 register function selection bit */
#define         bfd1_trdmr              trdmr_addr.bit.b7   /* GRD1 register function selection bit */

/*-----------------------------------------------------
   Timer RD PWM mode register
-----------------------------------------------------*/
union   byte_def  trdpmr_addr;
#define         trdpmr          trdpmr_addr.byte

#define         pwmb0_trdpmr    trdpmr_addr.bit.b0  /* PWM mode of TRDIOB0 selection bit */
#define         pwmc0_trdpmr    trdpmr_addr.bit.b1  /* PWM mode of TRDIOC0 selection bit */
#define         pwmd0_trdpmr    trdpmr_addr.bit.b2  /* PWM mode of TRDIOD0 selection bit */
#define         pwmb1_trdpmr    trdpmr_addr.bit.b4  /* PWM mode of TRDIOB1 selection bit */
#define         pwmc1_trdpmr    trdpmr_addr.bit.b5  /* PWM mode of TRDIOC1 selection bit */
#define         pwmd1_trdpmr    trdpmr_addr.bit.b6  /* PWM mode of TRDIOD1 selection bit */

/*-----------------------------------------------------
   Timer RD function control register
-----------------------------------------------------*/
union   byte_def  trdfcr_addr;
#define         trdfcr          trdfcr_addr.byte

#define         cmd0_trdfcr             trdfcr_addr.bit.b0  /* Combination mode selection bit */
#define         cmd1_trdfcr             trdfcr_addr.bit.b1  /* Combination mode selection bit */
#define         ols0_trdfcr             trdfcr_addr.bit.b2  /* Normal-Phase output level selection bit */
#define         ols1_trdfcr             trdfcr_addr.bit.b3  /* Counter-Phase output level selection bit */
#define         adtrg_trdfcr    trdfcr_addr.bit.b4  /* A/D trigger enable bit */
#define         adeg_trdfcr             trdfcr_addr.bit.b5  /* A/D trigger edge selection bit */
#define         stclk_trdfcr    trdfcr_addr.bit.b6  /* External clock input selection bit */
#define         pwm3_trdfcr             trdfcr_addr.bit.b7  /* Complementary PWM mode (Pair) selection bit */

/*-----------------------------------------------------
   Timer RD output master enable register 1
-----------------------------------------------------*/
union   byte_def  trdoer1_addr;
#define         trdoer1         trdoer1_addr.byte

#define         ea0_trdoer1             trdoer1_addr.bit.b0 /* TRDIOA0 output disable bit */
#define         eb0_trdoer1             trdoer1_addr.bit.b1 /* TRDIOB0 output disable bit */
#define         ec0_trdoer1             trdoer1_addr.bit.b2 /* TRDIOC0 output disable bit */
#define         ed0_trdoer1             trdoer1_addr.bit.b3 /* TRDIOD0 output disable bit */
#define         ea1_trdoer1             trdoer1_addr.bit.b4 /* TRDIOA1 output disable bit */
#define         eb1_trdoer1             trdoer1_addr.bit.b5 /* TRDIOB1 output disable bit */
#define         ec1_trdoer1             trdoer1_addr.bit.b6 /* TRDIOC1 output disable bit */
#define         ed1_trdoer1             trdoer1_addr.bit.b7 /* TRDIOD1 output disable bit */

/*-----------------------------------------------------
   Timer RD output master enable register 2
-----------------------------------------------------*/
```

```
union    byte_def  trdoer2_addr;
#define           trdoer2          trdoer2_addr.byte

#define           pto_trdoer2              trdoer2_addr.bit.b7 /* INT0 of pulse output forced cutoff signal input
enabled bit */

/*-----------------------------------------------------
   Timer RD output control register
-----------------------------------------------------*/
union    byte_def  trdocr_addr;
#define           trdocr           trdocr_addr.byte

#define           toa0_trdocr              trdocr_addr.bit.b0   /* TRDIOA0 output level selection bit */
#define           tob0_trdocr              trdocr_addr.bit.b1   /* TRDIOB0 output level selection bit */
#define           toc0_trdocr              trdocr_addr.bit.b2   /* TRDIOC0 initial output level selection bit */
#define           tod0_trdocr              trdocr_addr.bit.b3   /* TRDIOD0 initial output level selection bit */
#define           toa1_trdocr              trdocr_addr.bit.b4   /* TRDIOA1 initial output level selection bit */
#define           tob1_trdocr              trdocr_addr.bit.b5   /* TRDIOB1 initial output level selection bit */
#define           toc1_trdocr              trdocr_addr.bit.b6   /* TRDIOC1 initial output level selection bit */
#define           tod1_trdocr              trdocr_addr.bit.b7   /* TRDIOD1 initial output level selection bit */

/*-----------------------------------------------------
   Timer RD digital filter function contorol register 0
-----------------------------------------------------*/
union    byte_def  trddf0_addr;
#define           trddf0           trddf0_addr.byte

#define           dfa_trddf0               trddf0_addr.bit.b0   /* TRDIOA pin digital filter function selection bit */
#define           dfb_trddf0               trddf0_addr.bit.b1   /* TRDIOB pin digital filter function selection bit */
#define           dfc_trddf0               trddf0_addr.bit.b2   /* TRDIOC pin digital filter function selection bit */
#define           dfd_trddf0               trddf0_addr.bit.b3   /* TRDIOD pin digital filter function selection bit */
#define           dfck0_trddf0     trddf0_addr.bit.b6   /* Clock selection bit for digital filter function */
#define           dfck1_trddf0     trddf0_addr.bit.b7   /* Clock selection bit for digital filter function */

/*-----------------------------------------------------
   Timer RD digital filter function contorol register 1
-----------------------------------------------------*/
union    byte_def  trddf1_addr;
#define           trddf1           trddf1_addr.byte

#define           dfa_trddf1               trddf1_addr.bit.b0   /* TRDIOA pin digital filter function selection bit */
#define           dfb_trddf1               trddf1_addr.bit.b1   /* TRDIOB pin digital filter function selection bit */
#define           dfc_trddf1               trddf1_addr.bit.b2   /* TRDIOC pin digital filter function selection bit */
#define           dfd_trddf1               trddf1_addr.bit.b3   /* TRDIOD pin digital filter function selection bit */
#define           dfck0_trddf1     trddf1_addr.bit.b6   /* Clock selection bit for digital filter function */
#define           dfck1_trddf1     trddf1_addr.bit.b7   /* Clock selection bit for digital filter function */

/*-----------------------------------------------------
   Timer RD control register 0
-----------------------------------------------------*/
union    byte_def  trdcr0_addr;
#define           trdcr0           trdcr0_addr.byte

#define           tck0_trdcr0              trdcr0_addr.bit.b0   /* Count source selection bit */
#define           tck1_trdcr0              trdcr0_addr.bit.b1   /* Count source selection bit */
#define           tck2_trdcr0              trdcr0_addr.bit.b2   /* Count source selection bit */
#define           ckeg0_trdcr0     trdcr0_addr.bit.b3   /* External clock edge selection bit */
#define           ckeg1_trdcr0     trdcr0_addr.bit.b4   /* External clock edge selection bit */
#define           cclr0_trdcr0     trdcr0_addr.bit.b5   /* TRDCNTi counter clear selection bit */
#define           cclr1_trdcr0     trdcr0_addr.bit.b6   /* TRDCNTi counter clear selection bit */
#define           cclr2_trdcr0     trdcr0_addr.bit.b7   /* TRDCNTi counter clear selection bit */
```

```
/*-----------------------------------------------------
   Timer RD I/O control register A0
-----------------------------------------------------*/
union    byte_def  trdiora0_addr;
#define           trdiora0   trdiora0_addr.byte

#define           ioa0_trdiora0    trdiora0_addr.bit.b0/* GRA control bit */
#define           ioa1_trdiora0    trdiora0_addr.bit.b1/* GRA control bit */
#define           ioa2_trdiora0    trdiora0_addr.bit.b2/* GRA mode selection bit */
#define           ioa3_trdiora0    trdiora0_addr.bit.b3/* Input capture selection bit */
#define           iob0_trdiora0    trdiora0_addr.bit.b4/* GRB control bit */
#define           iob1_trdiora0    trdiora0_addr.bit.b5/* GRB control bit */
#define           iob2_trdiora0    trdiora0_addr.bit.b6/* GRB mode selection bit */

/*-----------------------------------------------------
   Timer RD I/O control register C0
-----------------------------------------------------*/
union    byte_def  trdiorc0_addr;
#define           trdiorc0   trdiorc0_addr.byte

#define           ioc0_trdiorc0    trdiorc0_addr.bit.b0/* GRC control bit */
#define           ioc1_trdiorc0    trdiorc0_addr.bit.b1/* GRC control bit */
#define           ioc2_trdiorc0    trdiorc0_addr.bit.b2/* GRC mode selection bit */
#define           ioc3_trdiorc0    trdiorc0_addr.bit.b3/* GRC register function selection bit */
#define           iod0_trdiorc0    trdiorc0_addr.bit.b4/* GRD control bit */
#define           iod1_trdiorc0    trdiorc0_addr.bit.b5/* GRD control bit */
#define           iod2_trdiorc0    trdiorc0_addr.bit.b6/* GRD mode selection bit */
#define           iod3_trdiorc0    trdiorc0_addr.bit.b7/* GRD register function selection bit */

/*-----------------------------------------------------
   Timer RD status register 0
-----------------------------------------------------*/
union    byte_def  trdsr0_addr;
#define           trdsr0           trdsr0_addr.byte

#define           imfa_trdsr0              trdsr0_addr.bit.b0   /* Input capture / compare match flag A */
#define           imfb_trdsr0              trdsr0_addr.bit.b1   /* Input capture / compare match flag B */
#define           imfc_trdsr0              trdsr0_addr.bit.b2   /* Input capture / compare match flag C */
#define           imfd_trdsr0              trdsr0_addr.bit.b3   /* Input capture / compare match flag D */
#define           ovf_trdsr0               trdsr0_addr.bit.b4   /* Overflow flag */
#define           udf_trdsr0               trdsr0_addr.bit.b5   /* Underflow flag */

/*-----------------------------------------------------
   Timer RD interrupt enable register 0
-----------------------------------------------------*/
union    byte_def  trdier0_addr;
#define           trdier0          trdier0_addr.byte

#define           imiea_trdier0    trdier0_addr.bit.b0  /* Input capture / compare match interrupt enable bit A */
#define           imieb_trdier0    trdier0_addr.bit.b1  /* Input capture / compare match interrupt enable bit B */
#define           imiec_trdier0    trdier0_addr.bit.b2  /* Input capture / compare match interrupt enable bit C */
#define           imied_trdier0    trdier0_addr.bit.b3  /* Input capture / compare match interrupt enable bit D */
#define           ovie_trdier0     trdier0_addr.bit.b4  /* Overflow / underflow interrupt enable bit */

/*-----------------------------------------------------
   Timer RD PWM mode output level control register 0
-----------------------------------------------------*/
union    byte_def  trdpocr0_addr;
#define           trdpocr0   trdpocr0_addr.byte

#define           polb_trdpocr0    trdpocr0_addr.bit.b0        /* PWM mode output level control bit B */
#define           polc_trdpocr0    trdpocr0_addr.bit.b1        /* PWM mode output level control bit C */
```

```
#define         pold_trdpocr0   trdpocr0_addr.bit.b2     /* PWM mode output level control bit D */

/*-----------------------------------------------------
  Timer RD control register 1
-----------------------------------------------------*/
union   byte_def  trdcr1_addr;
#define         trdcr1          trdcr1_addr.byte

#define         tck0_trdcr1             trdcr1_addr.bit.b0   /* Count source selection bit */
#define         tck1_trdcr1             trdcr1_addr.bit.b1   /* Count source selection bit */
#define         tck2_trdcr1             trdcr1_addr.bit.b2   /* Count source selection bit */
#define         ckeg0_trdcr1    trdcr1_addr.bit.b3   /* External clock edge selection bit */
#define         ckeg1_trdcr1    trdcr1_addr.bit.b4   /* External clock edge selection bit */
#define         cclr0_trdcr1    trdcr1_addr.bit.b5   /* TRDCNTi counter clear selection bit */
#define         cclr1_trdcr1    trdcr1_addr.bit.b6   /* TRDCNTi counter clear selection bit */
#define         cclr2_trdcr1    trdcr1_addr.bit.b7   /* TRDCNTi counter clear selection bit */

/*-----------------------------------------------------
  Timer RD I/O control register A1
-----------------------------------------------------*/
union   byte_def  trdiora1_addr;
#define         trdiora1   trdiora1_addr.byte

#define         ioa0_trdiora1   trdiora1_addr.bit.b0/* GRA control bit */
#define         ioa1_trdiora1   trdiora1_addr.bit.b1/* GRA control bit */
#define         ioa2_trdiora1   trdiora1_addr.bit.b2/* GRA mode selection bit */
#define         ioa3_trdiora1   trdiora1_addr.bit.b3/* Input capture selection bit */
#define         iob0_trdiora1   trdiora1_addr.bit.b4/* GRB control bit */
#define         iob1_trdiora1   trdiora1_addr.bit.b5/* GRB control bit */
#define         iob2_trdiora1   trdiora1_addr.bit.b6/* GRB mode selection bit */

/*-----------------------------------------------------
  Timer RD I/O control register C1
-----------------------------------------------------*/
union   byte_def  trdiorc1_addr;
#define         trdiorc1   trdiorc1_addr.byte

#define         ioc0_trdiorc1   trdiorc1_addr.bit.b0 /* GRC control bit */
#define         ioc1_trdiorc1   trdiorc1_addr.bit.b1 /* GRC control bit */
#define         ioc2_trdiorc1   trdiorc1_addr.bit.b2 /* GRC mode selection bit */
#define         ioc3_trdiorc1   trdiorc1_addr.bit.b3 /* GRC register function selection bit */
#define         iod0_trdiorc1   trdiorc1_addr.bit.b4 /* GRD control bit */
#define         iod1_trdiorc1   trdiorc1_addr.bit.b5 /* GRD control bit */
#define         iod2_trdiorc1   trdiorc1_addr.bit.b6 /* GRD mode selection bit */
#define         iod3_trdiorc1   trdiorc1_addr.bit.b7 /* GRD register function selection bit */

/*-----------------------------------------------------
  Timer RD status register 1
-----------------------------------------------------*/
union   byte_def  trdsr1_addr;
#define         trdsr1          trdsr1_addr.byte

#define         imfa_trdsr1             trdsr1_addr.bit.b0   /* Input capture / compare match flag A */
#define         imfb_trdsr1             trdsr1_addr.bit.b1   /* Input capture / compare match flag B */
#define         imfc_trdsr1             trdsr1_addr.bit.b2   /* Input capture / compare match flag C */
#define         imfd_trdsr1             trdsr1_addr.bit.b3   /* Input capture / compare match flag D */
#define         ovf_trdsr1              trdsr1_addr.bit.b4   /* Overflow flag */
#define         udf_trdsr1              trdsr1_addr.bit.b5   /* Underflow flag */

/*-----------------------------------------------------
  Timer RD interrupt enable register 1
-----------------------------------------------------*/
union   byte_def  trdier1_addr;
#define         trdier1         trdier1_addr.byte

#define         imiea_trdier1   trdier1_addr.bit.b0  /* Input capture / compare match interrupt enable bit A */
#define         imieb_trdier1   trdier1_addr.bit.b1  /* Input capture / compare match interrupt enable bit B */
#define         imiec_trdier1   trdier1_addr.bit.b2  /* Input capture / compare match interrupt enable bit C */
#define         imied_trdier1   trdier1_addr.bit.b3  /* Input capture / compare match interrupt enable bit D */
#define         ovie_trdier1    trdier1_addr.bit.b4  /* Overflow / underflow interrupt enable bit */

/*-----------------------------------------------------
  Timer RD PWM mode output level control register 1
-----------------------------------------------------*/
union   byte_def  trdpocr1_addr;
#define         trdpocr1   trdpocr1_addr.byte

#define         polb_trdpocr1   trdpocr1_addr.bit.b0     /* PWM mode output level control bit B */
#define         polc_trdpocr1   trdpocr1_addr.bit.b1     /* PWM mode output level control bit C */
#define         pold_trdpocr1   trdpocr1_addr.bit.b2     /* PWM mode output level control bit D */

/*-----------------------------------------------------
  UART1 bit rate register
-----------------------------------------------------*/
union   byte_def  u1brg_addr;
#define         u1brg           u1brg_addr.byte

/*-----------------------------------------------------
  Timer RA function select register
-----------------------------------------------------*/
union   byte_def  trasr_addr;
#define         trasr           trasr_addr.byte

#define         traiosel0       trasr_addr.bit.b0    /* TRAIO pin select bit */
#define         traiosel1       trasr_addr.bit.b1    /* TRAIO pin select bit */
#define         traosel0        trasr_addr.bit.b3    /* TRAO pin select bit */
#define         traosel1        trasr_addr.bit.b4    /* TRAO pin select bit */

/*-----------------------------------------------------
  Timer RB/RC   function select register
-----------------------------------------------------*/
union   byte_def  trbrcsr_addr;
#define         trbrcsr         trbrcsr_addr.byte

#define         trbosel0        trbrcsr_addr.bit.b0  /* TRBO pin select bit */
#define         trbosel1        trbrcsr_addr.bit.b1  /* TRBO pin select bit */
#define         trcclksel0      trbrcsr_addr.bit.b4  /* TRCCLK pin select bit */
#define         trcclksel1      trbrcsr_addr.bit.b5  /* TRCCLK pin select bit */

/*-----------------------------------------------------
  Timer RC function select register 0
-----------------------------------------------------*/
union   byte_def  trcpsr0_addr;
#define         trcpsr0         trcpsr0_addr.byte

#define         trcioasel0      trcpsr0_addr.bit.b0  /* TRCIOA/TRCTRG pin select bit */
#define         trcioasel1      trcpsr0_addr.bit.b1  /* TRCIOA/TRCTRG pin select bit */
#define         trcioasel2      trcpsr0_addr.bit.b2  /* TRCIOA/TRCTRG pin select bit */
#define         trciobsel0      trcpsr0_addr.bit.b4  /* TRCIOB pin select bit */
#define         trciobsel1      trcpsr0_addr.bit.b5  /* TRCIOB pin select bit */
#define         trciobsel2      trcpsr0_addr.bit.b6  /* TRCIOB pin select bit */

/*-----------------------------------------------------
  Timer RC function select register 1
```

```
----------------------------------------------------*/
union      byte_def  trcpsr1_addr;
#define              trcpsr1              trcpsr1_addr.byte

#define              trciocsel0           trcpsr1_addr.bit.b0  /* TRCIOC pin select bit */
#define              trciocsel1           trcpsr1_addr.bit.b1  /* TRCIOC pin select bit */
#define              trciocsel2           trcpsr1_addr.bit.b2  /* TRCIOC pin select bit */
#define              trciodsel0           trcpsr1_addr.bit.b4  /* TRCIOD pin select bit */
#define              trciodsel1           trcpsr1_addr.bit.b5  /* TRCIOD pin select bit */
#define              trciodsel2           trcpsr1_addr.bit.b6  /* TRCIOD pin select bit */

/*----------------------------------------------------
   Timer RD function select register 0
----------------------------------------------------*/
union      byte_def  trdpsr0_addr;
#define              trdpsr0              trdpsr0_addr.byte

#define              trdioa0sel0                trdpsr0_addr.bit.b0 /* TRDIOA0/TRDCLK pin select bit */
#define              trdiob0sel0                trdpsr0_addr.bit.b2 /* TRDIOB0 pin select bit */
#define              trdiob0sel1                trdpsr0_addr.bit.b3 /* TRDIOB0 pin select bit */
#define              trdioc0sel0                trdpsr0_addr.bit.b4 /* TRDIOC0 pin select bit */
#define              trdioc0sel1                trdpsr0_addr.bit.b5 /* TRDIOC0 pin select bit */
#define              trdiod0sel0                trdpsr0_addr.bit.b6 /* TRDIOD0 pin select bit */

/*----------------------------------------------------
   Timer RD function select register 1
----------------------------------------------------*/
union      byte_def  trdpsr1_addr;
#define              trdpsr1              trdpsr1_addr.byte

#define              trdioa1sel0                trdpsr1_addr.bit.b0 /* TRDIOA1 pin select bit */
#define              trdiob1sel0                trdpsr1_addr.bit.b2 /* TRDIOB1 pin select bit */
#define              trdioc1sel0                trdpsr1_addr.bit.b4 /* TRDIOC1 pin select bit */
#define              trdioc1sel1                trdpsr1_addr.bit.b5 /* TRDIOC1 pin select bit */
#define              trdiod1sel0                trdpsr1_addr.bit.b6 /* TRDIOD1 pin select bit */
#define              trdiod1sel1                trdpsr1_addr.bit.b7 /* TRDIOD1 pin select bit */

/*----------------------------------------------------
   Timer pin select register
----------------------------------------------------*/
union      byte_def  timsr_addr;
#define              timsr                timsr_addr.byte

#define              treosel0             timsr_addr.bit.b0    /* TREO pin select bit */

/*----------------------------------------------------
   UART0 function select register
----------------------------------------------------*/
union      byte_def  u0sr_addr;
#define              u0sr                 u0sr_addr.byte

#define              txd0sel0             u0sr_addr.bit.b0     /* TXD0 pin select bit */
#define              rxd0sel0             u0sr_addr.bit.b2     /* RXD0 pin select bit */
#define              clk0sel0             u0sr_addr.bit.b4     /* CLK0 pin select bit */

/*----------------------------------------------------
   UART1 function select register
----------------------------------------------------*/
union      byte_def  u1sr_addr;
#define              u1sr                 u1sr_addr.byte

#define              txd1sel0             u1sr_addr.bit.b0     /* TXD1 pin select bit */
#define              txd1sel1             u1sr_addr.bit.b1     /* TXD1 pin select bit */
#define              rxd1sel0             u1sr_addr.bit.b2     /* RXD1 pin select bit */
#define              rxd1sel1             u1sr_addr.bit.b3     /* RXD1 pin select bit */
#define              clk1sel0             u1sr_addr.bit.b4     /* CLK1 pin select bit */
#define              clk1sel1             u1sr_addr.bit.b5     /* CLK1 pin select bit */

/*----------------------------------------------------
   UART2 function select register 0
----------------------------------------------------*/
union      byte_def  u2sr0_addr;
#define              u2sr0                u2sr0_addr.byte

#define              txd2sel0             u2sr0_addr.bit.b0    /* TXD2/SDA2 pin select bit */
#define              txd2sel1             u2sr0_addr.bit.b1    /* TXD2/SDA2 pin select bit */
#define              txd2sel2             u2sr0_addr.bit.b2    /* TXD2/SDA2 pin select bit */
#define              rxd2sel0             u2sr0_addr.bit.b4    /* RXD2/SCK2 pin select bit */
#define              rxd2sel1             u2sr0_addr.bit.b5    /* RXD2/SCK2 pin select bit */
#define              rxd2sel2             u2sr0_addr.bit.b6    /* RXD2/SCK2 pin select bit */

/*----------------------------------------------------
   UART2 function select register 1
----------------------------------------------------*/
union      byte_def  u2sr1_addr;
#define              u2sr1                u2sr1_addr.byte

#define              clk2sel0             u2sr1_addr.bit.b0    /* CLK2 pin select bit */
#define              clk2sel1             u2sr1_addr.bit.b1    /* CLK2 pin select bit */
#define              cts2sel0             u2sr1_addr.bit.b4    /* CTS2/RTS2 pin select bit */
#define              cts2sel1             u2sr1_addr.bit.b5    /* CTS2/RTS2 pin select bit */

/*----------------------------------------------------
   UART2 function select register 1
----------------------------------------------------*/
union      byte_def  ssuiicsr_addr;
#define              ssuiicsr    ssuiicsr_addr.byte

#define              iicsel               ssuiicsr_addr.bit.b0 /* SSU/IIC bus change bit */
#define              sdasel0              ssuiicsr_addr.bit.b1 /* SDA pin select bit */
#define              ssisel0              ssuiicsr_addr.bit.b4 /* SSI pin select bit */
#define              ssisel1              ssuiicsr_addr.bit.b5 /* SSI pin select bit */
#define              scssel0              ssuiicsr_addr.bit.b6 /* SCS pin select bit */

/*----------------------------------------------------
   INT function select register
----------------------------------------------------*/
union      byte_def  intsr_addr;
#define              intsr                intsr_addr.byte

#define              int1sel0             intsr_addr.bit.b1    /* INT1 pin select bit */
#define              int1sel1             intsr_addr.bit.b2    /* INT1 pin select bit */
#define              int1sel2             intsr_addr.bit.b3    /* INT1 pin select bit */
#define              int2sel0             intsr_addr.bit.b4    /* INT2 pin select bit */
#define              int3sel0             intsr_addr.bit.b6    /* INT3 pin select bit */
#define              int3sel1             intsr_addr.bit.b7    /* INT3 pin select bit */

/*----------------------------------------------------
   INPUT/OUTPUT function select register
----------------------------------------------------*/
union      byte_def  pinsr_addr;
#define              pinsr                pinsr_addr.byte

#define              xcsel                          pinsr_addr.bit.b0    /* XCIN/XCOUT pin select bit */
```

```
/*-----------------------------------------------------
   SS bit counter register
-----------------------------------------------------*/
union      byte_def  ssbr_addr;
#define              ssbr                  ssbr_addr.byte

#define              bs0                                 ssbr_addr.bit.b0     /* SSU data transfer size set bit */
#define              bs1                                 ssbr_addr.bit.b1     /* SSU data transfer size set bit */
#define              bs2                                 ssbr_addr.bit.b2     /* SSU data transfer size set bit */
#define              bs3                                 ssbr_addr.bit.b3     /* SSU data transfer size set bit */

/*-----------------------------------------------------
   SS transmit data register
-----------------------------------------------------*/
union      byte_def  sstdr_addr;
#define              sstdr                 sstdr_addr.byte

/*-----------------------------------------------------
   IIC bus transmit data register
-----------------------------------------------------*/
union      byte_def  icdrt_addr;
#define              icdrt                 icdrt_addr.byte

/*-----------------------------------------------------
   SS transmit data register High
-----------------------------------------------------*/
union      byte_def  sstdrh_addr;
#define              sstdrh                sstdrh_addr.byte

/*-----------------------------------------------------
   SS receive data register
-----------------------------------------------------*/
union      byte_def  ssrdr_addr;
#define              ssrdr                 ssrdr_addr.byte

/*-----------------------------------------------------
   IIC bus receive data register
-----------------------------------------------------*/
union      byte_def  icdrr_addr;
#define              icdrr                 icdrr_addr.byte

/*-----------------------------------------------------
   SS receive data register High
-----------------------------------------------------*/
union      byte_def  ssrdrh_addr;
#define              ssrdrh                ssrdrh_addr.byte

/*-----------------------------------------------------
   SS control register H
-----------------------------------------------------*/
union      byte_def  sscrh_addr;
#define              sscrh                 sscrh_addr.byte

#define              cks0_sscrh                  sscrh_addr.bit.b0    /* Transfer clock rate select bit */
#define              cks1_sscrh                  sscrh_addr.bit.b1    /* Transfer clock rate select bit */
#define              cks2_sscrh                  sscrh_addr.bit.b2    /* Transfer clock rate select bit */
#define              mss_sscrh                   sscrh_addr.bit.b5    /* Master/Slave device select bit */
#define              rsstp_sscrh                 sscrh_addr.bit.b6    /* Receive single stop bit */

/*-----------------------------------------------------
   IIC bus control register 1
-----------------------------------------------------*/
union      byte_def  iccr1_addr;
#define              iccr1                 iccr1_addr.byte

#define              cks0_iccr1                  iccr1_addr.bit.b0    /* Transmit clock select bit 3 to 0 */
#define              cks1_iccr1                  iccr1_addr.bit.b1    /* Transmit clock select bit 3 to 0 */
#define              cks2_iccr1                  iccr1_addr.bit.b2    /* Transmit clock select bit 3 to 0 */
#define              cks3_iccr1                  iccr1_addr.bit.b3    /* Transmit clock select bit 3 to 0 */
#define              trs_iccr1             iccr1_addr.bit.b4    /* Transfer/receive select bit */
#define              mst_iccr1             iccr1_addr.bit.b5    /* Master/slave select bit */
#define              rcvd_iccr1                  iccr1_addr.bit.b6    /* Receive disable bit */
#define              ice_iccr1             iccr1_addr.bit.b7    /* IIC bus interface 2A enable bit */

/*-----------------------------------------------------
   SS control register L
-----------------------------------------------------*/
union      byte_def  sscrl_addr;
#define              sscrl                 sscrl_addr.byte

#define              sres_sscrl            sscrl_addr.bit.b1    /* SSUA control part reset bit */
#define              solp_sscrl            sscrl_addr.bit.b4    /* SOL write protect bit */
#define              sol_sscrl             sscrl_addr.bit.b5    /* Serial data output value setting bit */

/*-----------------------------------------------------
   IIC bus control register 2
-----------------------------------------------------*/
union      byte_def  iccr2_addr;
#define              iccr2                 iccr2_addr.byte

#define              iicrst_iccr2          iccr2_addr.bit.b1    /* IIC control part reset bit */
#define              scl0_iccr2            iccr2_addr.bit.b3    /* SCL monitor flag */
#define              sdaop_iccr2                 iccr2_addr.bit.b4    /* SDAO write protect bit */
#define              sdao_iccr2                  iccr2_addr.bit.b5    /* SDA output value control bit */
#define              scp_iccr2             iccr2_addr.bit.b6    /* Start/Stop condition generation disable bit */
#define              bbsy_iccr2                  iccr2_addr.bit.b7    /* Bus busy bit */

/*-----------------------------------------------------
   SS mode register
-----------------------------------------------------*/
union      byte_def  ssmr_addr;
#define              ssmr                  ssmr_addr.byte

#define              bc0_ssmr              ssmr_addr.bit.b0     /* Bit counter 2 to 0*/
#define              bc1_ssmr              ssmr_addr.bit.b1     /* Bit counter 2 to 0*/
#define              bc2_ssmr              ssmr_addr.bit.b2     /* Bit counter 2 to 0*/
#define              bc3_ssmr              ssmr_addr.bit.b3     /* Bit counter 2 to 0*/
#define              cphs_ssmr                   ssmr_addr.bit.b5     /* SSCK clock phase select bit */
#define              cpos_ssmr                   ssmr_addr.bit.b6     /* SSCK clock polarity select bit */
#define              mls_ssmr              ssmr_addr.bit.b7     /* MSB first/ LSB first select bit */

/*-----------------------------------------------------
   IIC bus mode register
-----------------------------------------------------*/
union      byte_def  icmr_addr;
#define              icmr                  icmr_addr.byte

#define              bc0_icmr              icmr_addr.bit.b0     /* Bit counter 2 to 0 */
#define              bc1_icmr              icmr_addr.bit.b1     /* Bit Counter 2 to 0 */
#define              bc2_icmr              icmr_addr.bit.b2     /* Bit Counter 2 to 0 */
#define              bcwp_icmr                   icmr_addr.bit.b3     /* BC write protect bit */
#define              wait_icmr             icmr_addr.bit.b6     /* Wait insertion bit */
#define              mls_icmr              icmr_addr.bit.b7     /* MSB-First/LSB-First select */
```

```
/*-----------------------------------------------------
   SS enable register
-----------------------------------------------------*/
union      byte_def  sser_addr;
#define             sser                sser_addr.byte

#define             ceie_sser           sser_addr.bit.b0    /* Conflict error interrupt enable bit */
#define             re_sser                 sser_addr.bit.b3    /* Receive enable bit */
#define             te_sser                 sser_addr.bit.b4    /* Transmit enable bit */
#define             rie_sser            sser_addr.bit.b5    /* Receive interrupt enable bit */
#define             teie_sser           sser_addr.bit.b6    /* Transmit end interrupt enable bit */
#define             tie_sser            sser_addr.bit.b7    /* Transmit interrupt enable bit */

/*-----------------------------------------------------
   IIC bus interrupt enable register
-----------------------------------------------------*/
union      byte_def  icier_addr;
#define             icier               icier_addr.byte

#define             ackbt_icier             icier_addr.bit.b0    /* Transmit acknow ledge select bit */
#define             ackbr_icier             icier_addr.bit.b1    /* Receive acknow ledge bit */
#define             acke_icier          icier_addr.bit.b2    /* Acknow ledge bit Judgement Select Bit */
#define             stie_icier          icier_addr.bit.b3    /* Stop condition detection interrupt enable bit */
#define             nakie_icier             icier_addr.bit.b4    /* NACK receive interrupt enable bit */
#define             rie_icier           icier_addr.bit.b5    /* Receive interrupt enable bit */
#define             teie_icier          icier_addr.bit.b6    /* Transmit end interrupt enable bit */
#define             tie_icier           icier_addr.bit.b7    /* Transmit interrupt enable bit */

/*-----------------------------------------------------
   SS status register
-----------------------------------------------------*/
union      byte_def  sssr_addr;
#define             sssr                sssr_addr.byte

#define             ce_sssr                 sssr_addr.bit.b0    /* Conflict error flag */
#define             orer_sssr           sssr_addr.bit.b2    /* Overrun error flag */
#define             rdrf_sssr           sssr_addr.bit.b5    /* Receive data register ful */
#define             tend_sssr           sssr_addr.bit.b6    /* Transmit end */
#define             tdre_sssr           sssr_addr.bit.b7    /* Transmit data empty */

/*-----------------------------------------------------
   IIC bus status register
-----------------------------------------------------*/
union      byte_def  icsr_addr;
#define             icsr                icsr_addr.byte

#define             adz_icsr            icsr_addr.bit.b0    /* General call address recognition flag */
#define             aas_icsr            icsr_addr.bit.b1    /* Slave address recognition flag */
#define             al_icsr                 icsr_addr.bit.b2    /* Arbitration lost flag */
#define             stop_icsr           icsr_addr.bit.b3    /* Stop condition detection flag */
#define             nackf_icsr              icsr_addr.bit.b4    /* No acknow ledge detection flag */
#define             rdrf_icsr           icsr_addr.bit.b5    /* Receive data register full */
#define             tend_icsr           icsr_addr.bit.b6    /* Transmit end */
#define             tdre_icsr           icsr_addr.bit.b7    /* Transmit data empty */

/*-----------------------------------------------------
   SS mode register 2
-----------------------------------------------------*/
union      byte_def  ssmr2_addr;
#define             ssmr2               ssmr2_addr.byte

#define             ssums_ssmr2             ssmr2_addr.bit.b0   /* SSUA mode select bit */
#define             csos_ssmr2              ssmr2_addr.bit.b1   /* SCS pin open drain output select bit */
#define             soos_ssmr2              ssmr2_addr.bit.b2   /* SSO pin open drain output select bit */
#define             sckos_ssmr2             ssmr2_addr.bit.b3   /* SSCK pin open drain output select bit */
#define             css0_ssmr2              ssmr2_addr.bit.b4   /* SCS pin selsct bit */
#define             css1_ssmr2              ssmr2_addr.bit.b5   /* SCS pin select bit */
#define             scks_ssmr2              ssmr2_addr.bit.b6   /* SSCK pin select bit */
#define             bide_ssmr2              ssmr2_addr.bit.b7   /* Bidirectional mode enable bit */

/*-----------------------------------------------------
   Slave address register
-----------------------------------------------------*/
union      byte_def  sar_addr;
#define             sar                             sar_addr.byte

#define             fs_sar                          sar_addr.bit.b0             /* Format select bit */
#define             sva0_sar            sar_addr.bit.b1             /* Slave address 6 to 0 */
#define             sva1_sar            sar_addr.bit.b2             /* Slave address 6 to 0 */
#define             sva2_sar            sar_addr.bit.b3             /* Slave address 6 to 0 */
#define             sva3_sar            sar_addr.bit.b4             /* Slave address 6 to 0 */
#define             sva4_sar            sar_addr.bit.b5             /* Slave address 6 to 0 */
#define             sva5_sar            sar_addr.bit.b6             /* Slave address 6 to 0 */
#define             sva6_sar            sar_addr.bit.b7             /* Slave address 6 to 0 */

/*-----------------------------------------------------
   Flash memory status register
-----------------------------------------------------*/
union      byte_def  fst_addr;
#define             fst                             fst_addr.byte

#define             rdysti                          fst_addr.bit.b0             /* Flash ready status interrupt request flag
*/
#define             bsyaei                          fst_addr.bit.b1             /* Flash access error interrupt request flag */
#define             lbdata                          fst_addr.bit.b2             /* LBDATA monitor flag */
#define             fst4                            fst_addr.bit.b4             /* Program error status flag */
#define             fst5                            fst_addr.bit.b5             /* Erase error status flag */
#define             fst6                            fst_addr.bit.b6             /* Erase suspend status flag */
#define             fst7                            fst_addr.bit.b7             /* Ready/Busy status flag */

/*-----------------------------------------------------
   Flash mamory control register0
-----------------------------------------------------*/
union      byte_def  fmr0_addr;
#define             fmr0                fmr0_addr.byte

#define             fmr01                           fmr0_addr.bit.b1    /* CPU rewrite mode select bit */
#define             fmr02                           fmr0_addr.bit.b2    /* EW1 mode select bit */
#define             fmstp                           fmr0_addr.bit.b3    /* Flash memory stop bit */
#define             cmdrst                          fmr0_addr.bit.b4    /* Erase/Write sequence reset bit */
#define             cmderie                         fmr0_addr.bit.b5    /* Erase/Write error interrupt enable bit */
#define             bsyaeie                         fmr0_addr.bit.b6    /* Flash access error interrupt enable bit */
#define             rdystie                         fmr0_addr.bit.b7    /* Flash ready status interrupt enable bit */

/*-----------------------------------------------------
   Flash mamory control register1
-----------------------------------------------------*/
union      byte_def  fmr1_addr;
#define             fmr1                fmr1_addr.byte

#define             fmr13                           fmr1_addr.bit.b3    /* Lock bit disabled select bit */
#define             fmr14                           fmr1_addr.bit.b4    /* BlockA rewrite disable bit */
#define             fmr15                           fmr1_addr.bit.b5    /* BlockB rewrite disable bit */
```

```
#define         fmr16                   fmr1_addr.bit.b6    /* BlockC rewrite disable bit */
#define         fmr17                   fmr1_addr.bit.b7    /* BlockD rewrite disable bit */

/*-----------------------------------------------------
   Flash mamory control register1
-----------------------------------------------------*/
union   byte_def  fmr2_addr;
#define         fmr2            fmr2_addr.byte

#define         fmr20                   fmr2_addr.bit.b0    /* Erase suspend enable bit */
#define         fmr21                   fmr2_addr.bit.b1    /* Erase suspend request bit */
#define         fmr22                   fmr2_addr.bit.b2    /* Program suspend request bit */
#define         fmr27                   fmr2_addr.bit.b7    /* Low-Power consumption read mode enable bit */

/*-----------------------------------------------------
   Address match interrupt enable register 0
-----------------------------------------------------*/
union   byte_def  aier0_addr;
#define         aier0           aier0_addr.byte

#define         aier00                  aier0_addr.bit.b0   /* Address match interrupt enable register 0 */

/*-----------------------------------------------------
   Address match interrupt enable register 1
-----------------------------------------------------*/
union   byte_def  aier1_addr;
#define         aier1           aier1_addr.byte

#define         aier10                  aier1_addr.bit.b0   /* Address match interrupt enable register 1 */

/*-----------------------------------------------------
   Pull-up control register0
-----------------------------------------------------*/
union   byte_def  pur0_addr;
#define         pur0            pur0_addr.byte

#define         pu00                    pur0_addr.bit.b0    /* P00 to P03 pull-up */
#define         pu01                    pur0_addr.bit.b1    /* P04 to P07 pull-up */
#define         pu02                    pur0_addr.bit.b2    /* P10 to P13 pull-up */
#define         pu03                    pur0_addr.bit.b3    /* P14 to P17 pull-up */
#define         pu04                    pur0_addr.bit.b4    /* P20 to P23 pull-up */
#define         pu05                    pur0_addr.bit.b5    /* P24 to P27 pull-up */
#define         pu06                    pur0_addr.bit.b6    /* P30 to P33 pull-up */
#define         pu07                    pur0_addr.bit.b7    /* P34 to P37 pull-up */

/*-----------------------------------------------------
   Pull-up control register1
-----------------------------------------------------*/
union   byte_def  pur1_addr;
#define         pur1            pur1_addr.byte

#define         pu10                    pur1_addr.bit.b0    /* P43 pull-up */
#define         pu11                    pur1_addr.bit.b1    /* P44 to P47 pull-up */
#define         pu13                    pur1_addr.bit.b3    /* P56, P57 pull-up */
#define         pu14                    pur1_addr.bit.b4    /* P60 to P63 pull-up */
#define         pu15                    pur1_addr.bit.b5    /* P64 to P67 pull-up */

/*-----------------------------------------------------
   Port P1 drivability control register
-----------------------------------------------------*/
union   byte_def  p1drr_addr;
#define         p1drr           p1drr_addr.byte
```

```
#define         p1drr0                  p1drr_addr.bit.b0   /* P10 drive capacity */
#define         p1drr1                  p1drr_addr.bit.b1   /* P11 drive capacity */
#define         p1drr2                  p1drr_addr.bit.b2   /* P12 drive capacity */
#define         p1drr3                  p1drr_addr.bit.b3   /* P13 drive capacity */
#define         p1drr4                  p1drr_addr.bit.b4   /* P14 drive capacity */
#define         p1drr5                  p1drr_addr.bit.b5   /* P15 drive capacity */
#define         p1drr6                  p1drr_addr.bit.b6   /* P16 drive capacity */
#define         p1drr7                  p1drr_addr.bit.b7   /* P17 drive capacity */

/*-----------------------------------------------------
   Port P2 drive capacity control register
-----------------------------------------------------*/
union   byte_def  p2drr_addr;
#define         p2drr           p2drr_addr.byte

#define         p2drr0                  p2drr_addr.bit.b0   /* P20 capacity */
#define         p2drr1                  p2drr_addr.bit.b1   /* P21 capacity */
#define         p2drr2                  p2drr_addr.bit.b2   /* P22 capacity */
#define         p2drr3                  p2drr_addr.bit.b3   /* P23 capacity */
#define         p2drr4                  p2drr_addr.bit.b4   /* P24 capacity */
#define         p2drr5                  p2drr_addr.bit.b5   /* P25 capacity */
#define         p2drr6                  p2drr_addr.bit.b6   /* P26 capacity */
#define         p2drr7                  p2drr_addr.bit.b7   /* P27 capacity */

/*-----------------------------------------------------
   Drivability control register 0
-----------------------------------------------------*/
union   byte_def  drr0_addr;
#define         drr0            drr0_addr.byte

#define         drr00                   drr0_addr.bit.b0    /* P00 to P03 drive capacity */
#define         drr01                   drr0_addr.bit.b1    /* P04 to P07 drive capacity */
#define         drr06                   drr0_addr.bit.b6    /* P30 to P33 drive capacity */
#define         drr07                   drr0_addr.bit.b7    /* P34 to P37 drive capacity */

/*-----------------------------------------------------
   Drivability control register 1
-----------------------------------------------------*/
union   byte_def  drr1_addr;
#define         drr1            drr1_addr.byte

#define         drr10                   drr1_addr.bit.b0    /* P43 drive capacity */
#define         drr11                   drr1_addr.bit.b1    /* P44 to P47 drive capacity */
#define         drr13                   drr1_addr.bit.b3    /* P56, P57 drive capacity */
#define         drr14                   drr1_addr.bit.b4    /* P60 to P63 drive capacity */
#define         drr15                   drr1_addr.bit.b5    /* P64 to P67 drive capacity */

/*-----------------------------------------------------
   Input threshold control register 0
-----------------------------------------------------*/
union   byte_def  vlt0_addr;
#define         vlt0            vlt0_addr.byte

#define         vlt00                   vlt0_addr.bit.b0    /* P0 input threshold select bit */
#define         vlt01                   vlt0_addr.bit.b1    /* P0 input threshold select bit */
#define         vlt02                   vlt0_addr.bit.b2    /* P1 input threshold select bit */
#define         vlt03                   vlt0_addr.bit.b3    /* P1 input threshold select bit */
#define         vlt04                   vlt0_addr.bit.b4    /* P2 input threshold select bit */
#define         vlt05                   vlt0_addr.bit.b5    /* P2 input threshold select bit */
#define         vlt06                   vlt0_addr.bit.b6    /* P3 input threshold select bit */
#define         vlt07                   vlt0_addr.bit.b7    /* P3 input threshold select bit */
```

```
/*-----------------------------------------------------
   Input threshold control register 1
-----------------------------------------------------*/
union      byte_def  vlt1_addr;
#define             vlt1                vlt1_addr.byte

#define             vlt10                       vlt1_addr.bit.b0    /* P42 to P47 input threshold select bit */
#define             vlt11                       vlt1_addr.bit.b1    /* P42 to P47 input threshold select bit */
#define             vlt12                       vlt1_addr.bit.b2    /* P56, P57 input threshold select bit */
#define             vlt13                       vlt1_addr.bit.b3    /* P56, P57 input threshold select bit */
#define             vlt14                       vlt1_addr.bit.b4    /* P6 input threshold select bit */
#define             vlt15                       vlt1_addr.bit.b5    /* P6 input threshold select bit */

/*-----------------------------------------------------
   comparator B control register
-----------------------------------------------------*/
union      byte_def  intcmp_addr;
#define             intcmp              intcmp_addr.byte

#define             int1cp0                     intcmp_addr.bit.b0  /* Comparator B1 operation enable bit */
#define             int1cout            intcmp_addr.bit.b3  /* Comparator B1 monitor flag */
#define             int3cp0                     intcmp_addr.bit.b4  /* Comparator B3 operation enable bit */
#define             int3cout            intcmp_addr.bit.b7  /* Comparator B3 monitor flag */

/*-----------------------------------------------------
   External interrupt enable register
-----------------------------------------------------*/
union      byte_def  inten_addr;
#define             inten               inten_addr.byte

#define             int0en                      inten_addr.bit.b0   /* INT0 input enable bit */
#define             int0pl                      inten_addr.bit.b1   /* INT0 input polarity select bit */
#define             int1en                      inten_addr.bit.b2   /* INT1 input enable bit */
#define             int1pl                      inten_addr.bit.b3   /* INT1 input polarity select bit */
#define             int2en                      inten_addr.bit.b4   /* INT2 input enable bit */
#define             int2pl                      inten_addr.bit.b5   /* INT2 input polarity select bit */
#define             int3en                      inten_addr.bit.b6   /* INT3 input enable bit */
#define             int3pl                      inten_addr.bit.b7   /* INT3 input polarity select bit */

/*-----------------------------------------------------
   external input enable register 1
-----------------------------------------------------*/
union      byte_def  inten1_addr;
#define             inten1              inten1_addr.byte

#define             int4en                      inten1_addr.bit.b0  /* INT4 input enable bit */
#define             int4pl                      inten1_addr.bit.b1  /* INT4 input polarity select bit */

/*-----------------------------------------------------
   INT0 input filter select register
-----------------------------------------------------*/
union      byte_def  intf_addr;
#define             intf                intf_addr.byte

#define             int0f0                      intf_addr.bit.b0    /* INT0 input filter select bit */
#define             int0f1                      intf_addr.bit.b1    /* INT0 input filter select bit */
#define             int1f0                      intf_addr.bit.b2    /* INT1 input filter select bit */
#define             int1f1                      intf_addr.bit.b3    /* INT1 input filter select bit */
#define             int2f0                      intf_addr.bit.b4    /* INT2 input filter select bit */
#define             int2f1                      intf_addr.bit.b5    /* INT2 input filter select bit */
#define             int3f0                      intf_addr.bit.b6    /* INT3 input filter select bit */
#define             int3f1                      intf_addr.bit.b7    /* INT3 input filter select bit */

/*-----------------------------------------------------
   INT input filter select register 1
-----------------------------------------------------*/
union      byte_def  intf1_addr;
#define             intf1               intf1_addr.byte

#define             int4f0                      intf1_addr.bit.b0   /* INT4 input filter select bit */
#define             int4f1                      intf1_addr.bit.b1   /* INT4 input filter select bit */

/*-----------------------------------------------------
   Key input enable register
-----------------------------------------------------*/
union      byte_def  kien_addr;
#define             kien                kien_addr.byte

#define             ki0en                       kien_addr.bit.b0    /* KI0 input enable bit */
#define             ki0pl                       kien_addr.bit.b1    /* KI0 input polarity select bit */
#define             ki1en                       kien_addr.bit.b2    /* KI1 input enable bit */
#define             ki1pl                       kien_addr.bit.b3    /* KI1 input polarity select bit */
#define             ki2en                       kien_addr.bit.b4    /* KI2 input enable bit */
#define             ki2pl                       kien_addr.bit.b5    /* KI2 input polarity select bit */
#define             ki3en                       kien_addr.bit.b6    /* KI3 input enable bit */
#define             ki3pl                       kien_addr.bit.b7    /* KI3 input polarity select bit */

/*-----------------------------------------------------
   Timer RC counter register, General register A,B,C,D
-----------------------------------------------------*/
unsigned short trc_addr, trcgra_addr, trcgrb_addr, trcgrc_addr, trcgrd_addr;

#define             trc                         trc_addr            /* Timer RC counter register */
#define             trcgra              trcgra_addr         /* Timer RC general register A */
#define             trcgrb              trcgrb_addr         /* Timer RC general register B */
#define             trcgrc              trcgrc_addr         /* Timer RC general register C */
#define             trcgrd              trcgrd_addr         /* Timer RC general register D */

/*-----------------------------------------------------
   Timer RC counter register, Timer RD Counter , General register A,B,C,D
-----------------------------------------------------*/
unsigned short trd0_addr, trdgra0_addr, trdgrb0_addr, trdgrc0_addr, trdgrd0_addr;
unsigned short trd1_addr, trdgra1_addr, trdgrb1_addr, trdgrc1_addr, trdgrd1_addr;

#define             trd0                trd0_addr           /* Timer RD counter 0 */
#define             trdgra0             trdgra0_addr        /* Timer RD general register A0 */
#define             trdgrb0             trdgrb0_addr        /* Timer RD general register B0 */
#define             trdgrc0             trdgrc0_addr        /* Timer RD general register C0 */
#define             trdgrd0             trdgrd0_addr        /* Timer RD general register D0 */

#define             trd1                trd1_addr           /* Timer RD counter 1 */
#define             trdgra1             trdgra1_addr        /* Timer RD general register A1 */
#define             trdgrb1             trdgrb1_addr        /* Timer RD general register B1 */
#define             trdgrc1             trdgrc1_addr        /* Timer RD general register C1 */
#define             trdgrd1             trdgrd1_addr        /* Timer RD general register D1 */

/*-----------------------------------------------------
   Interrupt control register
-----------------------------------------------------*/
union{
        struct{
                char        ilvl0:1;                                    /* Interrupt priority level select bit
*/
```

```
                char    ilvl1:1;                                        /* Interrupt priority level select bit
*/
                char    ilvl2:1;                                        /* Interrupt priority level select bit
*/
                char    ir:1;                                           /* Interrupt request bit
*/
                char    pol:1;                                          /* Polarity select bit */
                char    b5:1;
                char    b6:1;
                char    b7:1;
        } bit;
        char    byte;
} fmrdyic_addr, int4ic_addr, trcic_addr, trd0ic_addr, trd1ic_addr, treic_addr,
  s2tic_addr, s2ric_addr, kupic_addr, adic_addr, ssuic_addr, iicic_addr,
  s0tic_addr, s0ric_addr, s1tic_addr, s1ric_addr, int2ic_addr, traic_addr,
  trbic_addr, int1ic_addr, int3ic_addr, int0ic_addr, u2bcnic_addr, vcmp1ic_addr, vcmp2ic_addr;

/*----------------------------------------------------
  Flash memory ready interupt control register
----------------------------------------------------*/
#define         fmrdyic         fmrdyic_addr.byte

#define         ilvl0_fmrdyic   fmrdyic_addr.bit.ilvl0
#define         ilvl1_fmrdyic   fmrdyic_addr.bit.ilvl1
#define         ilvl2_fmrdyic   fmrdyic_addr.bit.ilvl2
#define         ir_fmrdyic              fmrdyic_addr.bit.ir

/*----------------------------------------------------
  INT4 interrupt control register
----------------------------------------------------*/
#define         int4ic          int4ic_addr.byte

#define         ilvl0_int4ic    int4ic_addr.bit.ilvl0
#define         ilvl1_int4ic    int4ic_addr.bit.ilvl1
#define         ilvl2_int4ic    int4ic_addr.bit.ilvl2
#define         ir_int4ic       int4ic_addr.bit.ir
#define         pol_int4ic      int4ic_addr.bit.pol

/*----------------------------------------------------
  Timer RC interrupt control register
----------------------------------------------------*/
#define         trcic           trcic_addr.byte

#define         ilvl0_trcic     trcic_addr.bit.ilvl0
#define         ilvl1_trcic     trcic_addr.bit.ilvl1
#define         ilvl2_trcic     trcic_addr.bit.ilvl2
#define         ir_trcic        trcic_addr.bit.ir

/*----------------------------------------------------
  Timer RD 0   interrupt control register
----------------------------------------------------*/
#define         trd0ic          trd0ic_addr.byte

#define         ilvl0_trd0ic    trd0ic_addr.bit.ilvl0
#define         ilvl1_trd0ic    trd0ic_addr.bit.ilvl1
#define         ilvl2_trd0ic    trd0ic_addr.bit.ilvl2
#define         ir_trd0ic       trd0ic_addr.bit.ir

/*----------------------------------------------------
  Timer RD 1   interrupt control register
----------------------------------------------------*/
#define         trd1ic          trd1ic_addr.byte

#define         ilvl0_trd1ic    trd1ic_addr.bit.ilvl0
#define         ilvl1_trd1ic    trd1ic_addr.bit.ilvl1
#define         ilvl2_trd1ic    trd1ic_addr.bit.ilvl2
#define         ir_trd1ic       trd1ic_addr.bit.ir

/*----------------------------------------------------
  Timer RE   interrupt control register
----------------------------------------------------*/
#define         treic           treic_addr.byte

#define         ilvl0_treic     treic_addr.bit.ilvl0
#define         ilvl1_treic     treic_addr.bit.ilvl1
#define         ilvl2_treic     treic_addr.bit.ilvl2
#define         ir_treic        treic_addr.bit.ir

/*----------------------------------------------------
  UART2 transmit interrupt control register
----------------------------------------------------*/
#define         s2tic           s2tic_addr.byte

#define         ilvl0_s2tic             s2tic_addr.bit.ilvl0
#define         ilvl1_s2tic             s2tic_addr.bit.ilvl1
#define         ilvl2_s2tic             s2tic_addr.bit.ilvl2
#define         ir_s2tic        s2tic_addr.bit.ir

/*----------------------------------------------------
  UART2 receive interrupt control register
----------------------------------------------------*/
#define         s2ric           s2ric_addr.byte

#define         ilvl0_s2ric             s2ric_addr.bit.ilvl0
#define         ilvl1_s2ric             s2ric_addr.bit.ilvl1
#define         ilvl2_s2ric             s2ric_addr.bit.ilvl2
#define         ir_s2ric        s2ric_addr.bit.ir

/*----------------------------------------------------
  Key input interrupt control register
----------------------------------------------------*/
#define         kupic           kupic_addr.byte

#define         ilvl0_kupic             kupic_addr.bit.ilvl0
#define         ilvl1_kupic             kupic_addr.bit.ilvl1
#define         ilvl2_kupic             kupic_addr.bit.ilvl2
#define         ir_kupic        kupic_addr.bit.ir

/*----------------------------------------------------
  A-D interrupt control register
----------------------------------------------------*/
#define         adic            adic_addr.byte

#define         ilvl0_adic      adic_addr.bit.ilvl0
#define         ilvl1_adic      adic_addr.bit.ilvl1
#define         ilvl2_adic      adic_addr.bit.ilvl2
#define         ir_adic                 adic_addr.bit.ir

/*----------------------------------------------------
  SSU interrupt control register
----------------------------------------------------*/
#define         ssuic           ssuic_addr.byte

#define         ilvl0_ssuic     ssuic_addr.bit.ilvl0
```

```
#define     ilvl1_ssuic     ssuic_addr.bit.ilvl1
#define     ilvl2_ssuic     ssuic_addr.bit.ilvl2
#define     ir_ssuic        ssuic_addr.bit.ir

/*-----------------------------------------------------
  IIC interrupt control register
-----------------------------------------------------*/
#define     iicic           iicic_addr.byte

#define     ilvl0_iicic iicic_addr.bit.ilvl0
#define     ilvl1_iicic iicic_addr.bit.ilvl1
#define     ilvl2_iicic iicic_addr.bit.ilvl2
#define     ir_iicic        iicic_addr.bit.ir

/*-----------------------------------------------------
  UART0 transmit interrupt control register
-----------------------------------------------------*/
#define     s0tic           s0tic_addr.byte

#define     ilvl0_s0tic             s0tic_addr.bit.ilvl0
#define     ilvl1_s0tic             s0tic_addr.bit.ilvl1
#define     ilvl2_s0tic             s0tic_addr.bit.ilvl2
#define     ir_s0tic        s0tic_addr.bit.ir

/*-----------------------------------------------------
  UART0 receive interrupt control register
-----------------------------------------------------*/
#define     s0ric           s0ric_addr.byte

#define     ilvl0_s0ric             s0ric_addr.bit.ilvl0
#define     ilvl1_s0ric             s0ric_addr.bit.ilvl1
#define     ilvl2_s0ric             s0ric_addr.bit.ilvl2
#define     ir_s0ric        s0ric_addr.bit.ir

/*-----------------------------------------------------
  UART1 transmit interrupt control register
-----------------------------------------------------*/
#define     s1tic           s1tic_addr.byte

#define     ilvl0_s1tic             s1tic_addr.bit.ilvl0
#define     ilvl1_s1tic             s1tic_addr.bit.ilvl1
#define     ilvl2_s1tic             s1tic_addr.bit.ilvl2
#define     ir_s1tic        s1tic_addr.bit.ir

/*-----------------------------------------------------
  UART1 receive interrupt control register
-----------------------------------------------------*/
#define     s1ric           s1ric_addr.byte

#define     ilvl0_s1ric             s1ric_addr.bit.ilvl0
#define     ilvl1_s1ric             s1ric_addr.bit.ilvl1
#define     ilvl2_s1ric             s1ric_addr.bit.ilvl2
#define     ir_s1ric        s1ric_addr.bit.ir

/*-----------------------------------------------------
  INT2 interrupt control register
-----------------------------------------------------*/
#define     int2ic          int2ic_addr.byte

#define     ilvl0_int2ic    int2ic_addr.bit.ilvl0
#define     ilvl1_int2ic    int2ic_addr.bit.ilvl1
#define     ilvl2_int2ic    int2ic_addr.bit.ilvl2
#define     ir_int2ic       int2ic_addr.bit.ir
#define     pol_int2ic      int2ic_addr.bit.pol

/*-----------------------------------------------------
  Timer RA interrupt control register
-----------------------------------------------------*/
#define     traic           traic_addr.byte

#define     ilvl0_traic             traic_addr.bit.ilvl0
#define     ilvl1_traic             traic_addr.bit.ilvl1
#define     ilvl2_traic             traic_addr.bit.ilvl2
#define     ir_traic        traic_addr.bit.ir

/*-----------------------------------------------------
  Timer RB interrupt control register
-----------------------------------------------------*/
#define     trbic           trbic_addr.byte

#define     ilvl0_trbic             trbic_addr.bit.ilvl0
#define     ilvl1_trbic             trbic_addr.bit.ilvl1
#define     ilvl2_trbic             trbic_addr.bit.ilvl2
#define     ir_trbic        trbic_addr.bit.ir

/*-----------------------------------------------------
  INT1 interrupt control register
-----------------------------------------------------*/
#define     int1ic          int1ic_addr.byte

#define     ilvl0_int1ic    int1ic_addr.bit.ilvl0
#define     ilvl1_int1ic    int1ic_addr.bit.ilvl1
#define     ilvl2_int1ic    int1ic_addr.bit.ilvl2
#define     ir_int1ic       int1ic_addr.bit.ir
#define     pol_int1ic      int1ic_addr.bit.pol

/*-----------------------------------------------------
  INT3 interrupt control register
-----------------------------------------------------*/
#define     int3ic          int3ic_addr.byte

#define     ilvl0_int3ic    int3ic_addr.bit.ilvl0
#define     ilvl1_int3ic    int3ic_addr.bit.ilvl1
#define     ilvl2_int3ic    int3ic_addr.bit.ilvl2
#define     ir_int3ic       int3ic_addr.bit.ir
#define     pol_int3ic      int3ic_addr.bit.pol

/*-----------------------------------------------------
  INT0 interrupt control register
-----------------------------------------------------*/
#define     int0ic          int0ic_addr.byte

#define     ilvl0_int0ic    int0ic_addr.bit.ilvl0
#define     ilvl1_int0ic    int0ic_addr.bit.ilvl1
#define     ilvl2_int0ic    int0ic_addr.bit.ilvl2
#define     ir_int0ic       int0ic_addr.bit.ir
#define     pol_int0ic      int0ic_addr.bit.pol

/*-----------------------------------------------------
  UART2 bus collision detection interrupt control register
-----------------------------------------------------*/
#define     u2bcnic         u2bcnic_addr.byte

#define     ilvl0_u2bcnic   u2bcnic_addr.bit.ilvl0
```

```
#define         ilvl1_u2bcnic       u2bcnic_addr.bit.ilvl1
#define         ilvl2_u2bcnic       u2bcnic_addr.bit.ilvl2
#define         ir_u2bcnic              u2bcnic_addr.bit.ir

/*-----------------------------------------------------
   UART2 bus collision detection interrupt control register
-----------------------------------------------------*/
#define         vcmp1ic             vcmp1ic_addr.byte

#define         ilvl0_vcmp1ic       vcmp1ic_addr.bit.ilvl0
#define         ilvl1_vcmp1ic       vcmp1ic_addr.bit.ilvl1
#define         ilvl2_vcmp1ic       vcmp1ic_addr.bit.ilvl2
#define         ir_vcmp1ic              vcmp1ic_addr.bit.ir

/*-----------------------------------------------------
   comparator A2 interrupt control register
-----------------------------------------------------*/
#define         vcmp2ic             vcmp2ic_addr.byte

#define         ilvl0_vcmp2ic       vcmp2ic_addr.bit.ilvl0
#define         ilvl1_vcmp2ic       vcmp2ic_addr.bit.ilvl1
#define         ilvl2_vcmp2ic       vcmp2ic_addr.bit.ilvl2
#define         ir_vcmp2ic              vcmp2ic_addr.bit.ir


/*-----------------------------------------------------
   UARTi transmit/receive mode register
-----------------------------------------------------*/
union{
        struct{
                char    smd0:1;                                 /* Serial I/O mode select
bit */
                char    smd1:1;                                 /* Serial I/O mode select
bit */
                char    smd2:1;                                 /* Serial I/O mode select
bit */
                char    ckdir:1;                        /* Internal/external clock select bit
*/
                char    stps:1;                                 /* Stop bit length select
bit */
                char    pry:1;                                  /* Odd/even parity select
bit */
                char    prye:1;                                 /* Parity enable bit */
                char    b7:1;
        } bit;
        char    byte;
} u0mr_addr, u1mr_addr, u2mr_addr;

/*-----------------------------------------------------
   UARTi transmit/receive mode register
-----------------------------------------------------*/
#define         u0mr            u0mr_addr.byte

#define         smd0_u0mr               u0mr_addr.bit.smd0      /* Serial I/O mode select bit */
#define         smd1_u0mr               u0mr_addr.bit.smd1      /* Serial I/O mode select bit */
#define         smd2_u0mr               u0mr_addr.bit.smd2      /* Serial I/O mode select bit */
#define         ckdir_u0mr              u0mr_addr.bit.ckdir/* Internal/external clock select bit */
#define         stps_u0mr               u0mr_addr.bit.stps  /* Stop bit length select bit */
#define         pry_u0mr        u0mr_addr.bit.pry   /* Odd/even parity select bit */
#define         prye_u0mr               u0mr_addr.bit.prye /* Parity enable bit */

#define         u1mr            u1mr_addr.byte

#define         smd0_u1mr               u1mr_addr.bit.smd0      /* Serial I/O mode select bit */
#define         smd1_u1mr               u1mr_addr.bit.smd1      /* Serial I/O mode select bit */
#define         smd2_u1mr               u1mr_addr.bit.smd2      /* Serial I/O mode select bit */
#define         ckdir_u1mr              u1mr_addr.bit.ckdir/* Internal/external clock select bit */
#define         stps_u1mr               u1mr_addr.bit.stps  /* Stop bit length select bit */
#define         pry_u1mr        u1mr_addr.bit.pry   /* Odd/even parity select bit */
#define         prye_u1mr               u1mr_addr.bit.prye /* Parity enable bit */

#define         u2mr            u2mr_addr.byte

#define         smd0_u2mr               u2mr_addr.bit.smd0      /* Serial I/O mode select bit */
#define         smd1_u2mr               u2mr_addr.bit.smd1      /* Serial I/O mode select bit */
#define         smd2_u2mr               u2mr_addr.bit.smd2      /* Serial I/O mode select bit */
#define         ckdir_u2mr              u2mr_addr.bit.ckdir/* Internal/external clock select bit */
#define         stps_u2mr               u2mr_addr.bit.stps  /* Stop bit length select bit */
#define         pry_u2mr        u2mr_addr.bit.pry   /* Odd/even parity select bit */
#define         prye_u2mr               u2mr_addr.bit.prye /* Parity enable bit */
#define         iopol_u2mr              u2mr_addr.bit.b7    /* TXD/RXD input/output polarity select bit */

/*-----------------------------------------------------
   UARTi transmit/receive control register0
-----------------------------------------------------*/
union{
        struct{
                char    clk0:1;                                 /*  BRG  count  source
select bit */
                char    clk1:1;                                 /*  BRG  count  source
select bit */
                char    crs:1;                                  /* CTS/RTS select bit */
                char    txept:1;                        /* Transmit register empty flag */
                char    crd:1;                                  /* CTS/RTS disable bit */
                char    nch:1;                                  /* Data output select bit
*/
                char    ckpol:1;                        /* CLK polarity select bit */
                char    uform:1;                         /* Transfer format select bit */
        } bit;
        char    byte;
} u0c0_addr, u1c0_addr, u2c0_addr;

/*-----------------------------------------------------
   UARTi transmit/receive control register0
-----------------------------------------------------*/
#define         u0c0            u0c0_addr.byte

#define         clk0_u0c0       u0c0_addr.bit.clk0  /* BRG count source select bit */
#define         clk1_u0c0       u0c0_addr.bit.clk1  /* BRG count source select bit */
#define         txept_u0c0              u0c0_addr.bit.txept /* Transmit register empty flag */
#define         nch_u0c0        u0c0_addr.bit.nch   /* Data output select bit */
#define         ckpol_u0c0              u0c0_addr.bit.ckpol /* CLK polarity select bit */
#define         uform_u0c0              u0c0_addr.bit.uform     /* Transfer format select bit */

#define         u1c0            u1c0_addr.byte

#define         clk0_u1c0       u1c0_addr.bit.clk0  /* BRG count source select bit */
#define         clk1_u1c0       u1c0_addr.bit.clk1  /* BRG count source select bit */
#define         txept_u1c0              u1c0_addr.bit.txept /* Transmit register empty flag */
#define         nch_u1c0        u1c0_addr.bit.nch   /* Data output select bit */
#define         ckpol_u1c0              u1c0_addr.bit.ckpol /* CLK polarity select bit */
#define         uform_u1c0              u1c0_addr.bit.uform     /* Transfer format select bit */

#define         u2c0            u2c0_addr.byte
```

```
#define         clk0_u2c0       u2c0_addr.bit.clk0   /* BRG count source select bit */
#define         clk1_u2c0       u2c0_addr.bit.clk1   /* BRG count source select bit */
#define         crs_u2c0        u2c0_addr.bit.crs    /* CTS/RTS select bit */
#define         txept_u2c0              u2c0_addr.bit.txept /* Transmit register empty flag */
#define         crd_u2c0        u2c0_addr.bit.crd    /* CTS/RTS disable bit */
#define         nch_u2c0        u2c0_addr.bit.nch    /* Data output select bit */
#define         ckpol_u2c0              u2c0_addr.bit.ckpol /* CLK polarity select bit */
#define         uform_u2c0              u2c0_addr.bit.uform         /* Transfer format select bit */

/*----------------------------------------------------
   UARTi transmit/receive control register1
----------------------------------------------------*/
union{
        struct{
                char    te:1;                                           /* Transmit enable bit */
                char    ti:1;                                           /* Transmit buffer empty
flag */
                char    re:1;                                           /* Receive enable bit */
                char    ri:1;                                           /* Receive complete flag
*/
                char    irs:1;
                char    rrm:1;
                char    lch:1;
                char    ere:1;
        } bit;
        char    byte;
} u0c1_addr, u1c1_addr, u2c1_addr;

/*----------------------------------------------------
   UARTi transmit/receive control register1
----------------------------------------------------*/
#define         u0c1            u0c1_addr.byte

#define         te_u0c1                 u0c1_addr.bit.te    /* Transmit enable bit */
#define         ti_u0c1                 u0c1_addr.bit.ti    /* Transmit buffer empty flag */
#define         re_u0c1                 u0c1_addr.bit.re    /* Receive enable bit */
#define         ri_u0c1                 u0c1_addr.bit.ri    /* Receive complete flag */
#define         u0irs_u0c1              u0c1_addr.bit.irs   /* UART0 transmit interrupt cause select bit */
#define         u0rrm_u0c1              u0c1_addr.bit.rrm   /* UART0 continuous receive mode enable bit */

#define         u1c1                    u1c1_addr.byte

#define         te_u1c1                 u1c1_addr.bit.te    /* Transmit enable bit */
#define         ti_u1c1                 u1c1_addr.bit.ti    /* Transmit buffer empty flag */
#define         re_u1c1                 u1c1_addr.bit.re    /* Receive enable bit */
#define         ri_u1c1                 u1c1_addr.bit.ri    /* Receive complete flag */
#define         u1irs_u1c1              u1c1_addr.bit.irs   /* UART1 transmit interrupt cause select bit */
#define         u1rrm_u1c1              u1c1_addr.bit.rrm   /* UART1 continuous receive mode enable bit */

#define         u2c1            u2c1_addr.byte

#define         te_u2c1                 u2c1_addr.bit.te    /* Transmit enable bit */
#define         ti_u2c1                 u2c1_addr.bit.ti    /* Transmit buffer empty flag */
#define         re_u2c1                 u2c1_addr.bit.re    /* Receive enable bit */
#define         ri_u2c1                 u2c1_addr.bit.ri    /* Receive complete flag */
#define         u2irs_u2c1              u2c1_addr.bit.irs   /* UART2 transmit interrupt cause select bit */
#define         u2rrm_u2c1              u2c1_addr.bit.rrm   /* UART2 continuous receive mode enable bit */
#define         u2lch_u2c1              u2c1_addr.bit.lch   /* Data output Logic select bit */
#define         u2ere_u2c1              u2c1_addr.bit.ere   /* Error signal output enabled bit */

/*----------------------------------------------------
   UARTi receive buffer register
----------------------------------------------------*/
union{
        struct{
                char    b0:1;
                char    b1:1;
                char    b2:1;
                char    b3:1;
                char    b4:1;
                char    b5:1;
                char    b6:1;
                char    b7:1;
                char    mprb:1;                                         /* Multiprocessor select
bit */
                char    b9:1;
                char    b10:1;
                char    abt:1;                                          /* Arbitration lost
detection flag */
                char    oer:1;                                          /* Overrun error flag */
                char    fer:1;                                          /* Framing error flag */
                char    per:1;                                          /* Parity error flag */
                char    sum:1;                                          /* Error sum flag */
        } bit;
        struct{
                char    low;                                            /* Low  8 bit */
                char    high;                                           /* High 8 bit */
        } byte;
        unsigned short  word;
} u0rb_addr, u1rb_addr, u2rb_addr;

/*----------------------------------------------------
   UARTi receive buffer register
----------------------------------------------------*/
#define         u0rb                    u0rb_addr.word
#define         u0rbl                   u0rb_addr.byte.low
#define         u0rbh                   u0rb_addr.byte.high

#define         oer_u0rb        u0rb_addr.bit.oer   /* Overrun error flag */
#define         fer_u0rb        u0rb_addr.bit.fer   /* Framing error flag */
#define         per_u0rb        u0rb_addr.bit.per   /* Parity error flag */
#define         sum_u0rb        u0rb_addr.bit.sum   /* Error sum flag */

#define         u1rb                    u1rb_addr.word
#define         u1rbl                   u1rb_addr.byte.low
#define         u1rbh                   u1rb_addr.byte.high

#define         oer_u1rb        u1rb_addr.bit.oer   /* Overrun error flag */
#define         fer_u1rb        u1rb_addr.bit.fer   /* Framing error flag */
#define         per_u1rb        u1rb_addr.bit.per   /* Parity error flag */
#define         sum_u1rb        u1rb_addr.bit.sum   /* Error sum flag */

#define         u2rb                    u2rb_addr.word
#define         u2rbl                   u2rb_addr.byte.low
#define         u2rbh                   u2rb_addr.byte.high

#define         mprb_u2ub               u2rb_addr.bit.mprb /* Multiprocessor select bit */
#define         abt_u2rb        u2rb_addr.bit.abt   /* Arbitration lost detection flag */
#define         oer_u2rb        u2rb_addr.bit.oer   /* Overrun error flag */
#define         fer_u2rb        u2rb_addr.bit.fer   /* Framing error flag */
#define         per_u2rb        u2rb_addr.bit.per   /* Parity error flag */
#define         sum_u2rb        u2rb_addr.bit.sum   /* Error sum flag */
```

```
/************************************************************
*    declare SFR union                                      *
************************************************************/

union{
        struct{
                char    b0:1;
                char    b1:1;
                char    b2:1;
                char    b3:1;
                char    b4:1;
                char    b5:1;
                char    b6:1;
                char    b7:1;
                char    b8:1;
                char    b9:1;
                char    b10:1;
                char    b11:1;
                char    b12:1;
                char    b13:1;
                char    b14:1;
                char    b15:1;
                char    b16:1;
                char    b17:1;
                char    b18:1;
                char    b19:1;
        } bit;
        struct{
                char    low;                                    /* low    8 bit */
                char    mid;                                    /* mid    8 bit */
                char    high;                                   /* high 8 bit */
                char    nc;                                             /* non use */
        } byte;

        unsigned long   dword;
} rmad0_addr,rmad1_addr;

#define         rmad0           rmad0_addr.dword        /* Address match interrupt register0 */
#define         rmad0l          rmad0_addr.byte.low             /* Address match interrupt register0 Low */
#define         rmad0m          rmad0_addr.byte.mid             /* Address match interrupt register0 Middle
*/
#define         rmad0h          rmad0_addr.byte.high    /* Address match interrupt register0 High */
#define         rmad1           rmad1_addr.dword        /* Address match interrupt register1 */
#define         rmad1l          rmad1_addr.byte.low             /* Address match interrupt register1 Low */
#define         rmad1m          rmad1_addr.byte.mid             /* Address match interrupt register1 Middle
*/
#define         rmad1h          rmad1_addr.byte.high    /* Address match interrupt register1 High */


union{
        struct{
                char    b0:1;
                char    b1:1;
                char    b2:1;
                char    b3:1;
                char    b4:1;
                char    b5:1;
                char    b6:1;
                char    b7:1;
                char    b8:1;
                char    b9:1;
                char    b10:1;
                char    b11:1;
                char    b12:1;
                char    b13:1;
                char    b14:1;
                char    b15:1;
        } bit;
        struct{
                char    low;                                    /* low    8 bit */
                char    high;                                   /* high 8 bit */
        } byte;
        unsigned int    word;
} u0tb_addr, u1tb_addr, u2tb_addr, ad0_addr, ad1_addr, ad2_addr, ad3_addr, ad4_addr, ad5_addr, ad6_addr, ad7_addr;

#define         u0tb            u0tb_addr.word                  /* UART0 transmit buffer register */
#define         u0tbl           u0tb_addr.byte.low      /* UART0 transmit buffer register Low */
#define         u0tbh           u0tb_addr.byte.high     /* UART0 transmit buffer register High */
#define         u1tb            u1tb_addr.word                  /* UART1 transmit buffer register */
#define         u1tbl           u1tb_addr.byte.low      /* UART1 transmit buffer register Low */
#define         u1tbh           u1tb_addr.byte.high     /* UART1 transmit buffer register High */
#define         u2tb            u2tb_addr.word                  /* UART2 transmit buffer register */
#define         u2tbl           u2tb_addr.byte.low      /* UART2 transmit buffer register Low */
#define         u2tbh           u2tb_addr.byte.high     /* UART2 transmit buffer register High */
#define         mptb_u2tb       u2tb_addr.bit.b8        /* UART2 MPTB    */
#define         ad0                     ad0_addr.word                   /* A-D register 0 */
#define         ad0l            ad0_addr.byte.low       /* A-D register 0 Low */
#define         ad0h            ad0_addr.byte.high      /* A-D register 0 High */
#define         ad1                     ad1_addr.word                   /* A-D register 1 */
#define         ad1l            ad1_addr.byte.low       /* A-D register 1 Low */
#define         ad1h            ad1_addr.byte.high      /* A-D register 1 High */
#define         ad2                     ad2_addr.word                   /* A-D register 2 */
#define         ad2l            ad2_addr.byte.low       /* A-D register 2 Low */
#define         ad2h            ad2_addr.byte.high      /* A-D register 2 High */
#define         ad3                     ad3_addr.word                   /* A-D register 3 */
#define         ad3l            ad3_addr.byte.low       /* A-D register 3 Low */
#define         ad3h            ad3_addr.byte.high      /* A-D register 3 High */
#define         ad4                     ad4_addr.word                   /* A-D register 4 */
#define         ad4l            ad4_addr.byte.low       /* A-D register 4 Low */
#define         ad4h            ad4_addr.byte.high      /* A-D register 4 High */
#define         ad5                     ad5_addr.word                   /* A-D register 5 */
#define         ad5l            ad5_addr.byte.low       /* A-D register 5 Low */
#define         ad5h            ad5_addr.byte.high      /* A-D register 5 High */
#define         ad6                     ad6_addr.word                   /* A-D register 6 */
#define         ad6l            ad6_addr.byte.low       /* A-D register 6 Low */
#define         ad6h            ad6_addr.byte.high      /* A-D register 6 High */
#define         ad7                     ad7_addr.word                   /* A-D register 7 */
#define         ad7l            ad7_addr.byte.low       /* A-D register 7 Low */
#define         ad7h            ad7_addr.byte.high      /* A-D register 7 High */
```