

I2C

- 16. I2C

16. I2C

I2C Inter-Integrated Circuit BUS I2C BUS IMX6ULL NXP P

16.1 I2C

16.1.1

I2C I2C

image-20220112152634031

Image not found or type unknown

I2C :

image-20220112152634031

Image not found or type unknown

①

- a. (start)
- b. A ()
- c.
- d. A
- e.

②

a. (start)

b. B ()

C. B

d. B

e.

IIC

- (start)
- (address)
- /
- / (ACK)
- IIC (P)

16.1.2

**** 1 ****

SDA(): SDA SDA SDA / device I2Cn_SI

```
**SCL(**      ) ** SDA
```

a. SDA SCL " "

b. " " " "

SDA CLK

image-20220112152634031

Image not found or type unknown

**** 2 ****

/

3

IIC

** 4 **

100kb/s, 400kb/s

** 5 **

16.1.2

1

I2C SDA SCL

image-20220112152634031

Image not found or type unknown

2

** ** SCL SDA I2C I2C

image-20220112152634031

Image not found or type unknown

3

8

4

SDA 8 transfer ACK MSB

image-20220112152634031

Image not found or type unknown

16.2 IMX6ULL I2C

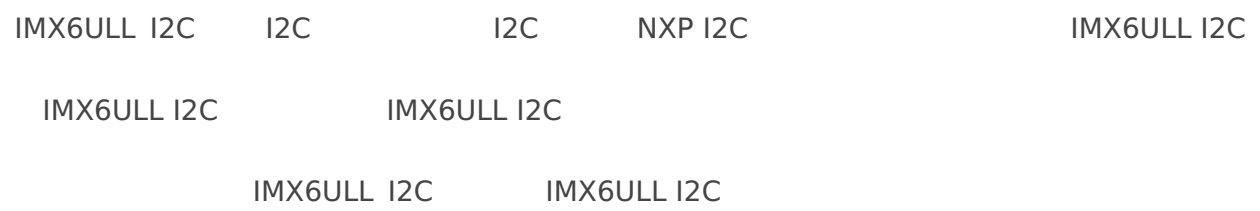
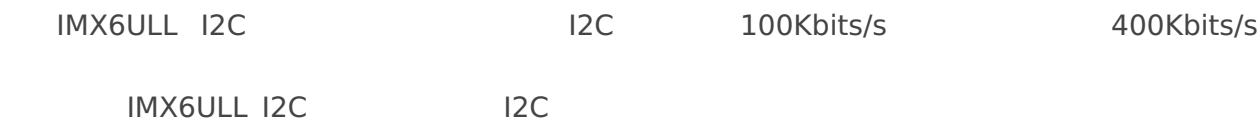


image-20220112152634031

Image not found or type unknown

IMX6ULL I2C



16.2.1 I2C Memory Map

I2C 5 16-bit

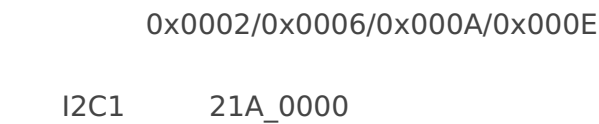


image-20220112152634031

Image not found or type unknown

16.2.2 Register

16.2.2.1 I2C Address Register (I2Cx_IADR) 0h

- a. bit15-8 0
- b. bit7-1 I2C I2C

16.2.2.2 I2C Frequency Divider Register (I2Cx_IFDR)

- a. bit15-6 0
- b. bit5-1 I2C

I2C_IFDR

IC :

I2CIPG_CLK_ROOT=66Mhz

PLL2 = 528 MHz

PLL2_PFD2 = 528 MHz

IPG_CLK_ROOT = (PLL2_PFD2 / ahb_podf)/ ipg_podf = (528 MHz/4)/2 = 66Mhz

PER_CLK_ROOT = IPG_CLK_ROOT/perclk_podf = 66 MHz/1 = 66 MHz

I2C 100K =66000000/100000=660.

Table 31-3. I2C_IFDR Register Field Values 640 0x15

image-20220112152634031

Image not found or type unknown

IFDR IC 0X15

16.2.2.3 I2C Control Register (I2Cx_I2CR) 8h

- a. bit 15-8 0

- b. **bit 7** I2C 0 disable 1 enable
- c. **bit 6** I2C 0 disable 1 enable
- d. **bit 5** / 0 slave mode 1 master mode
- e. **bit 4** 0 receive mode 1 transmit mode
- f. **bit 3** 0 ACK , 1 NO ACK
- g. **bit 2** 0 no repeat start 1 Generates repeat start
- h. **bit 0**

16.2.2.4 I2C Status Register (I2Cx_I2SR)

Ch

- a. **bit 15-8** 0
- b. **bit 7** 0 1
- c. **bit 6** I2C 0 1
- d. **bit 5** I2C 0 1
- e. **bit 4** 0 1
- f. **bit 3**
- g. **bit 2** 0 slave , 1 slave
- h. **bit 1** I2C 0 1
- i. **bit 0** 0 ACK 1 NO ACK

6.2.2.5 I2C Data I/O Register (I2Cx_I2DR)

10h

- a. **bit 15-8** 0
- b. **bit 7-1**

16.3 AP3216C

16.3.1 AP3216C

AP3216C

ALS: Ambient Light Sensor

PS: Proximity Sensor

16.3.2 AP3216C

- ① I2C
- ② FS mode 400kbit/s
- ③ ALS/PS+IR/ALS+PS+IR/PD/ALS once/SW Reset/PS+IR once/ALS+PS+IR once
- ④
- ⑤ -30°C to +80°C
- ⑥ ALS
 - a. (0-65536)16
 - b. 4
 - c. 50/60 HZ
 - d. @
 - e.
- ⑦ PS
 - a. (0-1023)10
 - b. 4
 - c.
 - d.
- ⑧ 4.1mm x 2.4mm x 1.35mm

16.3.3 AP3216C

AP3216C

image-20220112152634031

Image not found or type unknown

image-20220112152634031

Image not found or type unknown

AP3216C I2C

16.3.4 AP3216C

a. I2C

7bit /

AP3216 0X1E

b.

AP3216C 0x00 7bit

image-20220112152634031

Image not found or type unknown

AP3216C

16.4 I2C

Git NoosProgramProject/(16_I2C \001_example_i2c_ap3216c_led_show\i2c.c

Git NoosProgramProject/(16_I2C \001_example_i2c_ap3216c_led_show\i2c.h

I2C I2C1 I2C1

```
/*
 *The I2C contains five 16-bit registers.
 */
/*          (bit)          /
 * 21A_0000 (I2C1_IADR) 16   R/W 0000h 31. 7. 1/1463
 * 21A_0004 (I2C1_IFDR) 16   R/W 0000h 31. 7. 2/1463
```

```

* 21A_0008 (I2C1_I2CR) 16      R/W 0000h 31.7.3/1465
* 21A_000C (I2C1_I2SR) 16      R/W 0081h 31.7.4/1466
* 21A_0010 (I2C1_I2DR) 16      R/W 0000h 31.7.5/1468
*/
#define I2C1_BASE_ADDR          (0x21A0000u)
/* I2C1 Base address */
#define I2C1                    ((I2C_REGISTERS *)I2C1_BASE_ADDR)

```

I2C1-IC25

```

/*          */
/* all registers address is Base address + xh offset*/
typedef struct tagRegisters{
    volatile uint16_t IADR;          /*I2C Address Register, offset: 0x0 */
    uint8_t ReservedIADR[2];
    volatile uint16_t IFDR;          /*I2C Frequency Divider Register, offset: 0x4 */
    uint8_t ReservedIFDR[2];
    volatile uint16_t I2CR;          /*I2C Control Register, offset: 0x8 */
    uint8_t ReservedI2CR[2];
    volatile uint16_t I2SR;          /*I2C Status Register, offset: 0xC */
    uint8_t ReservedI2SR[2];
    volatile uint16_t I2DR;          /*I2C Data I/O Register, offset: 0x10 */
} I2C_REGISTERS;

```

```

typedef enum enI2C_OPCODE
{
    I2C_WRITE = 0,          /*          */
    I2C_READ  = 1,  []      /*          */
    I2C_DONOTHING_BULL
} I2C_OPCODE;

```

master

master

```

typedef struct tagI2cTransfer
{
    uint8_t ucSlaveAddress;    [] /* 7      */
    uint32_t ulOpcode ; []      /*      */
    uint32_t ulSubAddress;      /*      */
    uint8_t ulSubAddressLen;    [] /*      */

```

```
volatile uint32_t ullenth;  /* */
uint8_t *volatile pbuf;    /* */
} I2C_TRANSFER;
```

16.4.1 i2c_init

```
void i2c_init(I2C_REGISTERS *I2C_BASE); 81 void i2c_init(I2C_REGISTERS *I2C_BASE);
```

I2C1

I2CR(bit7)

image-20220112152634031

Image not found or type unknown

0 IFDR 0x15 100k 16.2 IMX6ULL I2C I2C I2CR(b

image-20220112152634031

Image not found or type unknown

```
I2C_BASE->I2CR &= ~(1 << 7);
I2C_BASE->IFDR = 0x15;
I2C_BASE->I2CR |= (1<<7);
```

16.4.2 i2c_transfer

```
uint8_t i2c_transfer(I2C_REGISTERS *I2C_BASE, I2C_TRANSFER *transfer);
```

I2C

image-20220112152634031

Image not found or type unknown

bit-4 IAL bit-1 IIF

image-20220112152634031

Image not found or type unknown

:

```
I2C_BASE->I2SR &= ~( (1 << 1) | (1 << 4));
```

ICF :

```
while(!((I2C_BASE->I2SR >> 7) & 0X1)){};
```

16.4.3 i2c_start

start I2C slave

```
uint8_t i2c_start(I2C_REGISTERS *I2C_BASE, uint8_t ucSlaveAddr, uint32_t ulOpcode);
```

IBB

```
if(I2C_BASE->I2SR & (1 << 5))
```

MSTA 1 master MTX 1 ,

image-20220112152634031

Image not found or type unknown

```
I2C_BASE->I2CR |= (1 << 5) | (1 << 4);
```

slave

```
63 I2C_BASE->I2DR = ((uint32_t)ucSlaveAddr << 1) | ((I2C_READ == ulOpcode)? 1 : 0);
```

16.4.4 i2c_check

check I2SR :

```
uint8_t i2c_check(I2C_REGISTERS *I2C_BASE, uint32_t status);
```

SR bit4 I2C SR bit0 1 NAK

```
uint8_t i2c_check(I2C_REGISTERS *I2C_BASE, uint32_t status);
{
    /*
    if(status & (1<<4))
    {
        I2C_BASE->I2SR &= ~(1<<4); /*
        /*

        I2C_BASE->I2CR &= ~(1 << 7); /* I2C */
        I2C_BASE->I2CR |= (1 << 7); /* I2C */
        return I2C_ARBITRATIONLOST;
    }
    else if(status & (1 << 0)) /*
    {
        return I2C_NAK; /* NAK(No acknowledge) */
    }
    return I2C_OK;
```

16.4.5 i2c_write

buf I2DR I2SR bit1 IIF ACK

```
void i2c_write(I2C_REGISTERS *I2C_BASE, const uint8_t *pbuf, uint32_t len)
{
    /*
    while(!(I2C_BASE->I2SR & (1 << 7)));

    I2C_BASE->I2SR &= ~(1 << 1); /*
    I2C_BASE->I2CR |= 1 << 4; /*
    while(len--)
    {
        I2C_BASE->I2DR = *pbuf++; /* buf I2DR */

        while(!(I2C_BASE->I2SR & (1 << 1))); /*
        I2C_BASE->I2SR &= ~(1 << 1); /*
```

```

    /* ACK */
    if(i2c_check( I2C_BASE, I2C_BASE->I2SR))
        break;

}

I2C_BASE->I2SR &= ~(1 << 1);
i2c_stop( I2C_BASE); /* */

}

```

16.4.6 i2c_read

<https://community.nxp.com/thread/378405>

```

void i2c_read( I2C_REGISTERS *I2C_BASE, uint8_t *pbuf, uint32_t len)
{
    volatile uint8_t dummy = 0;
    dummy++; /* */
    /* */
    while(!( I2C_BASE->I2SR & (1 << 7)));

    I2C_BASE->I2SR &= ~(1 << 1); /* */
    I2C_BASE->I2CR &= ~((1 << 4) | (1 << 3)); /* */

    /* NACK */
    if(len == 1)
        I2C_BASE->I2CR |= (1 << 3);
    dummy = I2C_BASE->I2DR; /* */
    while(len--)
    {
        while(!( I2C_BASE->I2SR & (1 << 1))); /* */
        I2C_BASE->I2SR &= ~(1 << 1); /* */
        if(len == 0)
        {
            i2c_stop( I2C_BASE); /* */
        }
        if(len == 1)
        {

```

```

        I2C_BASE->I2CR |= (1 << 3);
    }
    *pbuf++ = I2C_BASE->I2DR;
}
}

```

16.4.7 I2C

I2C

image-20220112152634031

Image not found or type unknown

- ① 1.Master START
- ② 2.Master I2C addr 7bit w 0 1bit ACK
- ③ 3.Slave ACK
- ④ 4.Master reg addr 8bit ACK
- ⑤ Slave ACK
- ⑥ Master data 8bit ACK
- ⑦ Slave ACK
- ⑧ 6 7
- ⑨ Master STOP

image-20220112152634031

Image not found or type unknown

- ① Master I2C addr 7bit w 1 1bit ACK
- ② Slave ACK
- ③ Master reg addr 8bit ACK

④ Slave ACK

⑤ Master RESTART

⑥ Master I2C addr 7bit r 1 1bit ACK

⑦ Slave ACK

⑧ Slave data 8bit

⑨ Master ACK

⑩ 8 9

Master NO ACK ACK

Master STOP

1.5 I2C _

I2C

****IEN:****I2C_I2CR bit6 (0 disable I2C interrupt 1 enable I2C interrupt)

****IIF:****I2C_I2SR (bit1) 0 No I2C interrupt pending 1 An interrupt is pending

1byte IIF ACK

16.6 AP3216C

AP3216C I2C write/read

Git **NoosProgramProject/(16_I2C \001_example_i2c_ap3216c_led_show\ap3216c.h)**
NoosProgramProject/(16_I2C \001_example_i2c_ap3216c_led_show\ap3216c.h)

16.6.1 AP3216C IO

100ASK_IMX6ULL

image-20220112152634031

Image not found or type unknown

image-20220112152634031

Image not found or type unknown

16.6.2 AP3216C

1.3-1.3.4	AP3216C	0X00	AP3216C
①	0X00	0X04	
②	0X03, ALS+PS+IR,	16.3	
③	()		

```
ret = i2c_write_one_byte( AP3216C_ADDR, AP3216C_SYSTEMCONG, 0X4);
mdelay 10 [ ]/* AP33216C 10ms */
ret = i2c_write_one_byte( AP3216C_ADDR, AP3216C_SYSTEMCONG, 0X3);
```

data 0X3

```
data = i2c_read_one_byte( AP3216C_ADDR, AP3216C_SYSTEMCONG);
```

i2c_write_one_byte i2c_read_one_byte i2c_transfer i2c.c

16.6.3 AP3216C

```
void ap3216c_read_data(uint16_t *ir, uint16_t *ps, uint16_t *als)
```

AP3216C_IRDATALOW AP3216C_PSDATAHIGH 6 6 ir ps als

```
void ap3216c_read_data(uint16_t *ir, uint16_t *ps, uint16_t *als)
{
```

```

uint8_t buf[ 6];
uint8_t i;
for(i = 0; i < 6; i++) /*
{
    buf[ i] = i2c_read_one_byte( AP3216C_ADDR, AP3216C_IRDATALOW + i);
}
if(buf[ 0] & 0X80) /* IR_OF 1,
{
    *ir = 0;
}
else /* IR
{
    *ir = ((uint16_t)buf[ 1] << 2) | (buf[ 0] & 0X03);
}
/* ALS
*als = ((uint16_t)buf[ 3] << 8) | buf[ 2];

if(buf[ 4] & 0x40) /* IR_OF 1,
{
    *ps = 0;
}
else /* PS
{
    *ps = ((uint16_t)(buf[ 5] & 0X3F) << 4) | (buf[ 4] & 0X0F);
}
return;
}

```

0X0A~0X0F 6

ALS PS IR

ALS PS IR

112.5ms

image-20220112152634031

Image not found or type unknown

16.7 AP3216C

AP3216C I2C

IMX6ULL

image-20220112152634031

Image not found or type unknown

①

②

③

④

16.7.1 AP3216C

Git NoosProgramProject/(16_I2C \001_example_i2c_ap3216c_led_show)

1-3

```
while(!ret)
{
    /*      (ALS)      (PS)      (IR) */
    ap3216c_read_data(&ir,&ps,&als);
    /*          LED */
    if (als>100 || ps >1000)
    {
        led_ctl(1);
    }
    else
    {
        led_ctl(0);
    }
}
```

16.7.2 4-1.4

Git NoosProgramProject/(16_I2C \001_example_i2c_ap3216c_led_show)

16.7.3 3-1.4

LED ALS PS

image-20220112152634031

Image not found or type unknown

image-20220112152634031

Image not found or type unknown

16.7.4 AP3216C

Git NoosProgramProject/(16_I2C \002_example_i2c_ap3216c_printf_show)

1-3

```
while(1)
{
    delay(100000);
    /*      (ALS)      (PS)      (IR) */
    ap3216c_read_data(&ir, &ps, &als);

    printf("ir=%d ps=%d als=%d\n\r", ir, ps, als);
}
```

16.7.5 4-1.4

****** Git NoosProgramProject/(16_I2C \002_example_i2c_ap3216c_printf_show)******

16.7.6 3-1.4

image-20220112152634031

Image not found or type unknown