MSP430FR5969 UART 测试

实例代码：

```
//  MSP430FR59xx Demo - USCI_A0 External Loopback test @ 115200 baud
//
//  Description: This demo connects TX to RX of the MSP430 UART
//  The example code shows proper initialization of registers
//  and interrupts to receive and transmit data. If data is incorrect P1.0 LED is
//  turned ON.
//  ACLK = n/a, MCLK = SMCLK = BRCLK = default DCO = 1MHz
//
//
//               MSP430FR5969
//            -----------------
//       RST -|     P2.0/UCA0TXD|----|
//            |                 |    |
//           -|                 |    |
//            |     P2.1/UCA0RXD|----|
//            |                 |
//            |             P1.0|---> LED
//
//   P. Thanigai
//   Texas Instruments Inc.
//   August 2012
//   Built with CCS V4 and IAR Embedded Workbench Version: 5.5
//******************************************************************************
#include <msp430.h>

volatile unsigned char RXData = 0;
volatile unsigned char TXData = 1;

int main(void)
{
  WDTCTL = WDTPW | WDTHOLD;                 // Stop watchdog

  // Configure GPIO
  P1OUT &= ~BIT0;                           // Clear P1.0 output latch
  P1DIR |= BIT0;                            // For LED on P1.0
  P2SEL1 |= BIT0 | BIT1;                    // USCI_A0 UART operation
  P2SEL0 &= ~(BIT0 | BIT1);

  // Disable the GPIO power-on default high-impedance mode to activate
  // previously configured port settings
  PM5CTL0 &= ~LOCKLPM5;

  // Configure USCI_A0 for UART mode
  UCA0CTLW0 = UCSWRST;                      // Put eUSCI in reset
  UCA0CTL1 |= UCSSEL__SMCLK;                // CLK = SMCLK
  UCA0BR0 = 8;                              // 1000000/115200 = 8.68
  UCA0MCTLW = 0xD600;                       // 1000000/115200 - INT(1000000/115200)=0.68
                                            // UCBRSx value = 0xD6 (See UG)
  UCA0BR1 = 0;
  UCA0CTL1 &= ~UCSWRST;                     // release from reset
  UCA0IE |= UCRXIE;                         // Enable USCI_A0 RX interrupt

  while (1)
  {
    while(!(UCA0IFG & UCTXIFG));
    UCA0TXBUF = TXData++;                   // Load data onto buffer

  }
}

#if defined(__TI_COMPILER_VERSION__) || defined(__IAR_SYSTEMS_ICC__)
#pragma vector=USCI_A0_VECTOR
__interrupt void USCI_A0_ISR(void)
#elif defined(__GNUC__)
void __attribute__ ((interrupt(USCI_A0_VECTOR))) USCI_A0_ISR (void)
#else
#error Compiler not supported!
#endif
{
  switch(__even_in_range(UCA0IV,USCI_UART_UCTXCPTIFG))
  {
    case USCI_NONE: break;
    case USCI_UART_UCRXIFG:
      RXData = UCA0RXBUF;                   // Read buffer
      if(RXData != TXData)                  // Check value
      {
        P1OUT |= BIT0;                      // If incorrect turn on P1.0
          while(1);                         // Trap CPU
```

```
    }
    TXData++;                       // increment data byte

    break;
  case USCI_UART_UCTXIFG: break;
  case USCI_UART_UCSTTIFG: break;
  case USCI_UART_UCTXCPTIFG: break;
 }
}
```

分析：
1、对于MSP430单片机，如果是要验证程序的话，首先需要关闭看门狗定时器，以保证正常调试。在产品量产后，可以不关闭看门狗来防止系统死机，但需要计算程序执行时间，进行喂狗操作。

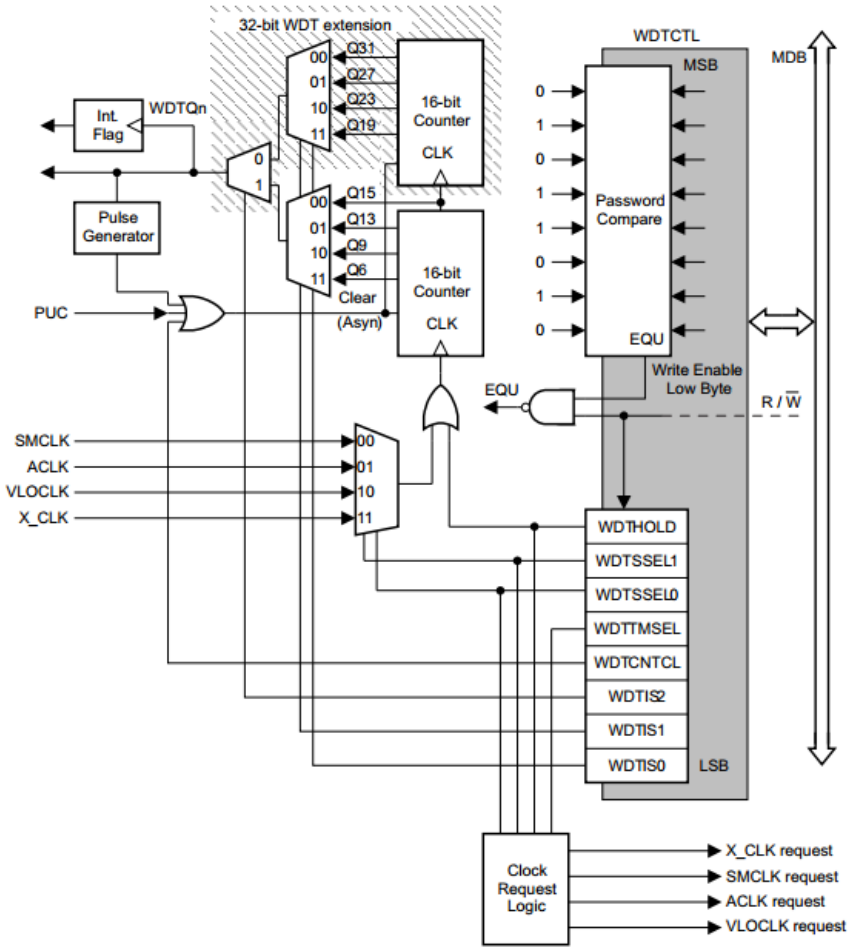


Figure 18-1. Watchdog Timer Block Diagram

通过以下代码关闭看门狗。

```
WDTCTL = WDTPW | WDTHOLD;
```

其中：WDTPW = 0x5A00, WDTHOLD = 0x0080. 写入密码并关闭看门狗。

Figure 18-2. WDTCTL Register

15	14	13	12	11	10	9	8
WDTPW							
rw	rw	rw	rw	rw	rw	rw	rw
7	6	5	4	3	2	1	0
WDTHOLD	WDTSSEL		WDTTMSEL	WDTCNTCL	WDTIS		
rw-0	rw-0	rw-0	rw-0	r0(w)	rw-1	rw-0	rw-0

Table 18-2. WDTCTL Register Description

Bit	Field	Type	Reset	Description
15-8	WDTPW	RW	69h	Watchdog timer password. Always read as 069h. Must be written as 05Ah, or a PUC is generated.
7	WDTHOLD	RW	0h	Watchdog timer hold. This bit stops the watchdog timer. Setting WDTHOLD = 1 when the WDT is not in use conserves power. 0b = Watchdog timer is not stopped 1b = Watchdog timer is stopped
6-5	WDTSSEL	RW	0h	Watchdog timer clock source select 00b = SMCLK 01b = ACLK 10b = VLOCLK 11b = X_CLK, same as VLOCLK if not defined differently in data sheet
4	WDTTMSEL	RW	0h	Watchdog timer mode select 0b = Watchdog mode 1b = Interval timer mode
3	WDTCNTCL	RW	0h	Watchdog timer counter clear. Setting WDTCNTCL = 1 clears the count value to 0000h. WDTCNTCL is automatically reset. 0b = No action 1b = WDTCNT = 0000h
2-0	WDTIS	RW	4h	Watchdog timer interval select. These bits select the watchdog timer interval to set the WDTIFG flag or generate a PUC. 000b = Watchdog clock source / 2^{31} (18:12:16 at 32.768 kHz) 001b = Watchdog clock source / 2^{27} (01:08:16 at 32.768 kHz) 010b = Watchdog clock source / 2^{23} (00:04:16 at 32.768 kHz) 011b = Watchdog clock source / 2^{19} (00:00:16 at 32.768 kHz) 100b = Watchdog clock source / 2^{15} (1 s at 32.768 kHz) 101b = Watchdog clock source / 2^{13} (250 ms at 32.768 kHz) 110b = Watchdog clock source / 2^{9} (15.625 ms at 32.768 kHz) 111b = Watchdog clock source / 2^{6} (1.95 ms at 32.768 kHz)

2、配置GPIO口。

```
// Configure GPIO
P1OUT &= ~BIT0;                         // Clear P1.0 output latch
P1DIR |= BIT0;                          // For LED on P1.0
```

通过下面的文档说明可以知道如何配置：

12.2 Digital I/O Operation

The digital I/O are configured with user software. The setup and operation of the digital I/O are discussed in the following sections.

12.2.1 Input Registers (PxIN)

Each bit in each PxIN register reflects the value of the input signal at the corresponding I/O pin when the pin is configured as I/O function. These registers are read only.

- Bit = 0: Input is low
- Bit = 1: Input is high

NOTE: Writing to read-only registers PxIN

Writing to these read-only registers results in increased current consumption while the write attempt is active.

12.2.2 Output Registers (PxOUT)

Each bit in each PxOUT register is the value to be output on the corresponding I/O pin when the pin is configured as I/O function, output direction.

- Bit = 0: Output is low
- Bit = 1: Output is high

If the pin is configured as I/O function, input direction and the pullup or pulldown resistor are enabled; the corresponding bit in the PxOUT register selects pullup or pulldown.

- Bit = 0: Pin is pulled down
- Bit = 1: Pin is pulled up

12.2.3 Direction Registers (PxDIR)

Each bit in each PxDIR register selects the direction of the corresponding I/O pin, regardless of the selected function for the pin. PxDIR bits for I/O pins that are selected for other functions must be set as required by the other function.

- Bit = 0: Port pin is switched to input direction
- Bit = 1: Port pin is switched to output direction

12.2.4 Pullup or Pulldown Resistor Enable Registers (PxREN)

Each bit in each PxREN register enables or disables the pullup or pulldown resistor of the corresponding I/O pin. The corresponding bit in the PxOUT register selects if the pin contains a pullup or pulldown.

- Bit = 0: Pullup or pulldown resistor disabled
- Bit = 1: Pullup or pulldown resistor enabled

Table 12-1 summarizes the use of PxDIR, PxREN, and PxOUT for proper I/O configuration.

Table 12-1. I/O Configuration

PxDIR	PxREN	PxOUT	I/O Configuration
0	0	x	Input
0	1	0	Input with pulldown resistor
0	1	1	Input with pullup resistor
1	x	x	Output

3、设置引脚为外设模式。

```
P2SEL1 |= BIT0 | BIT1;                  // USCI_A0 UART operation
P2SEL0 &= ~(BIT0 | BIT1);
```

通过下面的文档说明可以知道如何配置：

Table 12-2. I/O Function Selection

PxSEL1	PxSEL0	I/O Function
0	0	General purpose I/O is selected
0	1	Primary module function is selected
1	0	Secondary module function is selected
1	1	Tertiary module function is selected

因为PxSEL1和PxSEL0位不驻留在连续的地址中，所以不能同时更改这两个位。

4、解锁端口配置。

```
PM5CTL0 &= ~LOCKLPM5;
```

其中，LOCKLPM5 = 0x0001.可参考下面的文档说明

2.3.4 PM5CTL0 Register (offset = 10h) [reset = 0001h]

Power Mode 5 Control Register 0

Figure 2-7. PM5CTL0 Register

15	14	13	12	11	10	9	8
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
Reserved							LOCKLPM5
r0	r0	r0	r0	r0	r0	r0	rw-{1}

Table 2-5. PM5CTL0 Register Description

Bit	Field	Type	Reset	Description
15-1	Reserved	R	0h	Reserved. Always reads as 0.
0	LOCKLPM5	RW	1h	Locks I/O pin and other LPMx.5 relevant (for example, RTC) configurations upon exit from LPMx.5. This bit is set by hardware and must be cleared by software. It cannot be set by software. After a power cycle I/O pins are locked in high-impedance state with input Schmitt triggers disabled until LOCKLPM5 is cleared by the user software. After a wake-up from LPMx.5 I/O pins and other LPMx.5 relevant (for example, RTC) configurations are locked in their states configured before LPMx.5 entry until LOCKLPM5 is cleared by the user software. 0b = I/O pin and LPMx.5 configurations unlocked. 1b = I/O pin and LPMx.5 configuration remains locked.

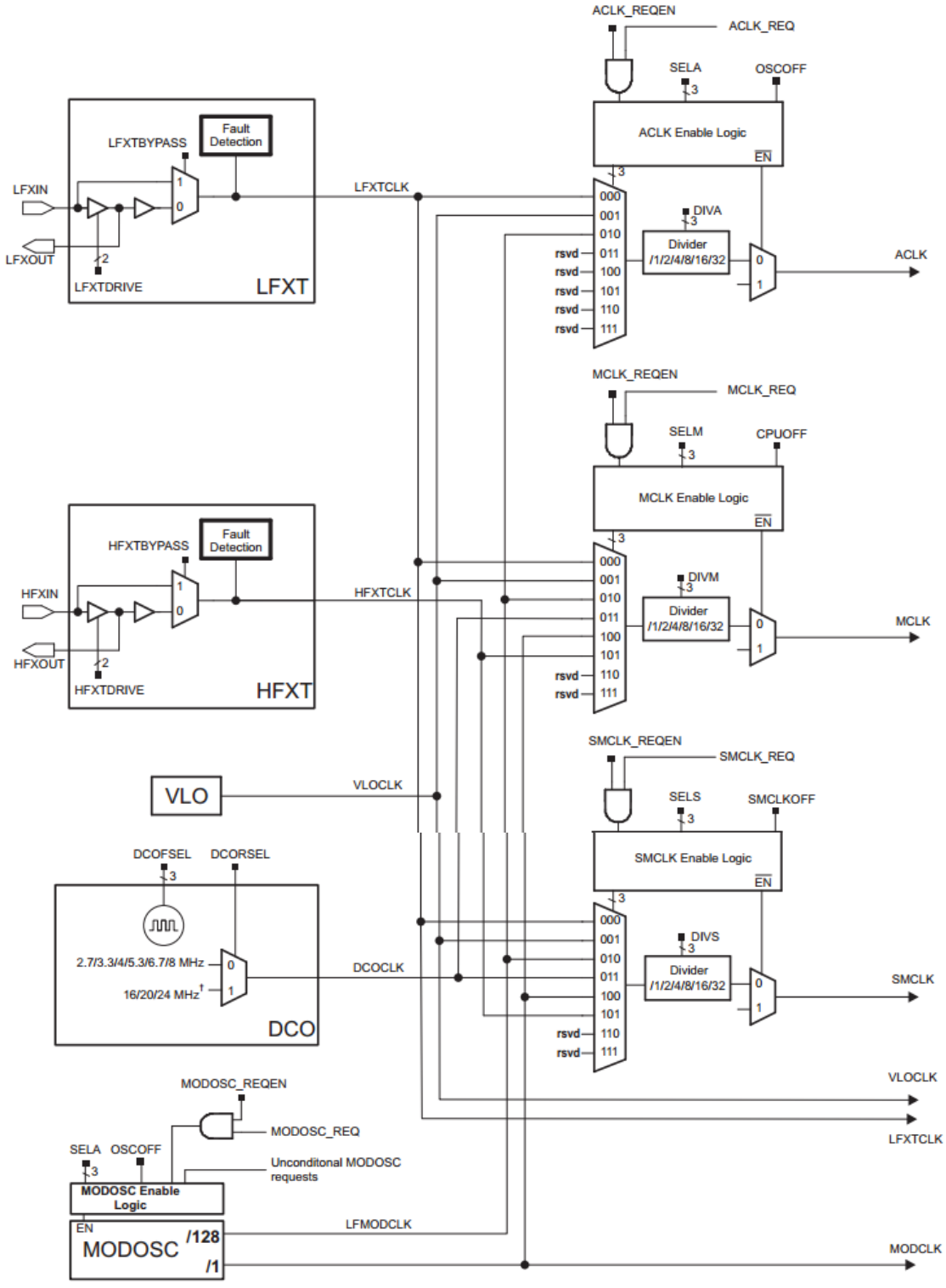
5、配置时钟。

这里没有配置时钟，为默认值：

```
//  ACLK = n/a, MCLK = SMCLK = BRCLK = default DCO = 1MHz
```

如果需要配置时钟，可以参考下面的配置方法：

The block diagram of the clock system module is shown in Figure 3-1.



† Not available on all devices

Figure 3-1. Clock System Block Diagram

时钟系统操作

在PUC之后，CS模块默认配置为：

•选择LFXT作为LFXTCLK的振荡源。选择LFXTCLK为ACLK（SELAx = 0）的时钟源，ACLK不可分频（DIVAx = 0）。

•选择DCOCLK为MCLK和SMCLK（SELMx = SELSx = 3）时钟源，每个为8分频（DIVMx = DIVSx = 3）。

•LFXIN和LFXOUT引脚设置为通用I / O，并且LFXT保持禁用，直到I / O端口配置为LFXT工作模式。

•HFXIN和HFXOUT引脚设置为通用I / O，HFXT禁用。

如前所述，默认情况下选择LFXT，但LFXT被禁用。晶体管脚（LFXIN，LFXOUT）与通用I / O共用。要使能LFXT，必须设置与晶体管脚相关的PSEL位。当LFXTCLK使用32768-Hz晶体时，故障控制逻辑立即导致时钟源切换，ACLK由LFMODCLK提供，MCLK和SMCLK由MODCLK提供，因为LFXT没有立即稳定（见第3.2.8节）。

状态寄存器控制位（SCG0，SCG1，OSCOFF和CPUOFF）配置MSP430工作模式，并启用或禁用时钟系统模块的部分（请参见系统复位，中断和 操作模式章节）。通过CSCTL6注册CSCTL0，配置CS模块。

在程序执行期间，CS模块可以随时由软件进行配置或重新配置。 CS控制寄存器受密码保护，以防止无意中访问。

实例说明：配置 ACLK = 32768Hz, MCLK = SMCLK = BRCLK = default DCO = ~1MHz

首先，我们配置LFXIN和LFXOUT为LFXT工作模式。

```
PJSEL0 |= BIT4 | BIT5;                  // Configure XT1 pins
```

注意，之后需要解锁端口。

接下来，设置时钟：

（1）解锁时钟，CS控制寄存器受密码保护，以防止无意中访问。

3.3.1 CSCTL0 Register

Clock System Control 0 Register

Figure 3-5. CSCTL0 Register

15	14	13	12	11	10	9	8
CSKEY							
rw-1	rw-0	rw-0	rw-1	rw-0	rw-1	rw-1	rw-0
7	6	5	4	3	2	1	0
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0

Table 3-4. CSCTL0 Register Description

Bit	Field	Type	Reset	Description
15-8	CSKEY	RW	96h	CSKEY password. Must always be written with A5h; a PUC is generated if any other value is written. Always reads as 96h. After the correct password is written, all CS registers are available for writing.
7-0	Reserved	R	0h	Reserved. Always reads as 0.

```
CSCTL0_H = CSKEY >> 8;                  // Unlock CS registers
```

其中，CSKEY = 0xA500。

（2）设置DCO时钟源频率。需要DCORESL和DCOFSEL配合设置。

3.3.2 CSCTL1 Register

Clock System Control 1 Register

Figure 3-6. CSCTL1 Register

15	14	13	12	11	10	9	8
Reserved							
r0	r0	r0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
Reserved	DCORSEL	Reserved		DCOFSEL			Reserved
r0	rw-[0]	r0	r0	rw-[1]	rw-[1]	rw-[0]	r0

Table 3-5. CSCTL1 Register Description

Bit	Field	Type	Reset	Description
15-7	Reserved	R	0h	Reserved. Always reads as 0.
6	DCORSEL	RW	0h	DCO range select. For high-speed devices, this bit can be written by the user. For low-speed devices, it is always 0. See description of DCOFSEL bit for details.
5-4	Reserved	R	0h	Reserved. Always reads as 0.
3-1	DCOFSEL	RW	6h	DCO frequency select. Selects frequency settings for the DCO. Values shown below are approximate. Refer to the device-specific data sheet. 000b = If DCORSEL = 0: 1 MHz; If DCORSEL = 1: 1 MHz 001b = If DCORSEL = 0: 2.67 MHz; If DCORSEL = 1: 5.33 MHz 010b = If DCORSEL = 0: 3.5 MHz; If DCORSEL = 1: 7 MHz 011b = If DCORSEL = 0: 4 MHz; If DCORSEL = 1: 8 MHz 100b = If DCORSEL = 0: 5.33 MHz; If DCORSEL = 1: 16 MHz 101b = If DCORSEL = 0: 7 MHz; If DCORSEL = 1: 21 MHz 110b = If DCORSEL = 0: 8 MHz; If DCORSEL = 1: 24 MHz 111b = If DCORSEL = 0: Reserved. Defaults to 8 MHz. It is not recommended to use this setting; If DCORSEL = 1: Reserved. Defaults to 24 MHz. It is not recommended to use this setting
0	Reserved	R	0h	Reserved. Always reads as 0.

```
CSCTL1 = DCOFSEL_0;                     // Set DCO to 1MHz
```

其中，DCOFSEL_0 = 0x0000。

（3）设置各时钟的时钟源

Table 3-6. CSCTL2 Register Description

Bit	Field	Type	Reset	Description
15-11	Reserved	R	0h	Reserved. Always reads as 0.
10-8	SELA	RW	0h	Selects the ACLK source 000b = LFXTCLK when LFXT available, otherwise VLOCLK. 001b = VLOCLK 010b = LFMODCLK 011b = Reserved. Defaults to LFMODCLK. Not recommended for use to ensure future compatibility. 100b = Reserved. Defaults to LFMODCLK. Not recommended for use to ensure future compatibility. 101b = Reserved. Defaults to LFMODCLK. Not recommended for use to ensure future compatibility. 110b = Reserved. Defaults to LFMODCLK. Not recommended for use to ensure future compatibility. 111b = Reserved. Defaults to LFMODCLK. Not recommended for use to ensure future compatibility.
7	Reserved	R	0h	Reserved. Always reads as 0.
6-4	SELS	RW	3h	Selects the SMCLK source 000b = LFXTCLK when LFXT available, otherwise VLOCLK. 001b = VLOCLK 010b = LFMODCLK 011b = DCOCLK 100b = MODCLK 101b = HFXTCLK when HFXT available, otherwise DCOCLK. 110b = Reserved. Defaults to HFXTCLK. Not recommended for use to ensure future compatibility. 111b = Reserved. Defaults to HFXTCLK. Not recommended for use to ensure future compatibility.
3	Reserved	R	0h	Reserved. Always reads as 0.
2-0	SELM	RW	3h	Selects the MCLK source 000b = LFXTCLK when LFXT available, otherwise VLOCLK 001b = VLOCLK 010b = LFMODCLK 011b = DCOCLK 100b = MODCLK 101b = HFXTCLK when HFXT available, otherwise DCOCLK 110b = Reserved. Defaults to HFXTCLK. Not recommended for use to ensure future compatibility. 111b = Reserved. Defaults to HFXTCLK. Not recommended for use to ensure future compatibility.

```
CSCTL2 = SELA__LFXTCLK | SELS__DCOCLK | SELM__DCOCLK;
```

其中，SELA__LFXTCLK = 0x0000,SELS__DCOCLK = 0x0030,SELM__DCOCLK = 0x0003。

（4）设置分频参数

Table 3-7. CSCTL3 Register Description

Bit	Field	Type	Reset	Description
15-11	Reserved	R	0h	Reserved. Always reads as 0.
10-8	DIVA	RW	0h	ACLK source divider. Divides the frequency of the ACLK clock source. 000b = /1 001b = /2 010b = /4 011b = /8 100b = /16 101b = /32 110b = Reserved. Defaults to /32. Not recommended for use to ensure future compatibility. 111b = Reserved. Defaults to /32. Not recommended for use to ensure future compatibility.
7	Reserved	R	0h	Reserved. Always reads as 0.
6-4	DIVS	RW	3h	SMCLK source divider. Divides the frequency of the SMCLK clock source. 000b = /1 001b = /2 010b = /4 011b = /8 100b = /16 101b = /32 110b = Reserved. Defaults to /32. Not recommended for use to ensure future compatibility. 111b = Reserved. Defaults to /32. Not recommended for use to ensure future compatibility.
3	Reserved	R	0h	Reserved. Always reads as 0.
2-0	DIVM	RW	3h	MCLK source divider. Divides the frequency of the MCLK clock source. 000b = /1 001b = /2 010b = /4 011b = /8 100b = /16 101b = /32 110b = Reserved. Defaults to /32. Not recommended for use to ensure future compatibility. 111b = Reserved. Defaults to /32. Not recommended for use to ensure future compatibility.

```
CSCTL3 = DIVA__1 | DIVS__1 | DIVM__1;     // Set all dividers
```

其中，DIVA_1、DIVS_1、DIVM_1 都为0x0000。

（5）开启LFXT

Table 3-8. CSCTL4 Register Description

Bit	Field	Type	Reset	Description
15-14	HFXTDRIVE	RW	3h	The HFXT oscillator current can be adjusted to its drive needs. This in combination with the HFFREQ bits can be used for optimizing crystal power based on crystal characteristics. 00b = Lowest current consumption 01b = Increased drive strength HFXT oscillator 10b = Increased drive strength HFXT oscillator 11b = Maximum drive strength HFXT oscillator
13	Reserved	R	0h	Reserved. Always reads as 0.
12	HFXTBYPASS	RW	0h	HFXT bypass select 0b = HFXT sourced from external crystal 1b = HFXT sourced from external clock signal
11-10	HFFREQ	RW	3h	The HFXT frequency selection. These bits must be set to the appropriate frequency for crystal or bypass modes of operation. 00b = 0 to 4 MHz 01b = Greater than 4 MHz to 8 MHz 10b = Greater than 8 MHz to 16 MHz 11b = Greater than 16 MHz to 24 MHz
9	Reserved	R	0h	Reserved. Always reads as 0.
8	HFXTOFF	RW	1h	Turns off the HFXT oscillator 0b = HFXT is on if HFXT is selected through the port selection and HFXT is not in bypass mode of operation 1b = HFXT is off if it is not used as a source for ACLK, MCLK, or SMCLK
7-6	LFXTDRIVE	RW	3h	The LFXT oscillator current can be adjusted to its drive needs. 00b = Lowest drive strength and current consumption LFXT oscillator 01b = Increased drive strength LFXT oscillator 10b = Increased drive strength LFXT oscillator 11b = Maximum drive strength and maximum current consumption LFXT oscillator
5	Reserved	RW	0h	Reserved. Must be written as zero.
4	LFXTBYPASS	RW	0h	LFXT bypass select 0b = LFXT sourced from external crystal 1b = LFXT sourced from external clock signal
3	VLOOFF	RW	1h	VLO off. This bit turns off the VLO. 0b = VLO is on 1b = VLO is off if it is not used as a source for ACLK, MCLK, or SMCLK or if not used as a source for the RTC in LPM3.5
2	Reserved	R	0h	Reserved. Always reads as 0.
1	SMCLKOFF	RW	0h	SMCLK off. This bit turns off the SMCLK. 0b = SMCLK on 1b = SMCLK off

Table 3-8. CSCTL4 Register Description (continued)

Bit	Field	Type	Reset	Description
0	LFXTOFF	RW	1h	LFXT off. This bit turns off the LFXT. 0b = LFXT is on if LFXT is selected through the port selection and LFXT is not in bypass mode of operation 1b = LFXT is off if it is not used as a source for ACLK, MCLK, or SMCLK

```
CSCTL4 &= ~LFXTOFF;                    // Enable LFXT1
```

其中，LFXTOFF = 0x0001。

这里有一个需要注意里面一个名词LFXTBYPASS的理解，之前使用STM32时，里面有一个说明，可以参考下：

外部时钟源(HSE旁路)

在这个模式里，必须提供外部时钟。它的频率最高可达25MHz。用户可通过设置在时钟控制寄存器中的HSEBYP和HSEON位来选择这一模式。外部时钟信号(50%占空比的方波、正弦波或三角波)必须连到SOC_IN管脚，同时保证OSC_OUT管脚悬空。见图8。

外部晶体/陶瓷谐振器(HSE晶体)

4~16Mz外部振荡器可为系统提供更为精确的主时钟。相关的硬件配置可参考图8，进一步信息可参考数据手册的电气特性部分。

在时钟控制寄存器RCC_CR中的HSERDY位用来指示高速外部振荡器是否稳定。在启动时，直到这一位被硬件置'1'，时钟才被释放出来。如果在时钟中断寄存器RCC_CIR中允许产生中断，将会产生相应中断。

HSE晶体可以通过设置时钟控制寄存器里RCC_CR中的HSEON位被启动和关闭。

（6）检测晶振是否正常

首先软件清除寄存器里对应的HFXT或者LFXT晶振错误标志。接着测试晶振是否正常。

3.3.6 CSCTL5 Register

Clock System Control 5 Register

Figure 3-10. CSCTL5 Register

15	14	13	12	11	10	9	8
Reserved							
r0	r0	0	r0	r0	r0	r0	r0
7	6	5	4	3	2	1	0
ENSTFCNT2	ENSTFCNT1	Reserved				HFXTOFFG	LFXTOFFG
rw-(1)	rw-(1)	r0	r0	r0	r0	rw-(0)	rw-(1)

Table 3-9. CSCTL5 Register Description

Bit	Field	Type	Reset	Description
15-8	Reserved	R	0h	Reserved. Always reads as 0.
7	ENSTFCNT2	RW	1h	Enable start counter for HFXT when available. 0b = Startup fault counter disabled. Counter is cleared. 1b = Startup fault counter enabled
6	ENSTFCNT1	RW	1h	Enable start counter for LFXT. 0b = Startup fault counter disabled. Counter is cleared. 1b = Startup fault counter enabled
5-2	Reserved	R	0h	Reserved. Always reads as 0.
1	HFXTOFFG	RW	0h	HFXT oscillator fault flag. If this bit is set, the OFIFG flag is also set. HFXTOFFG is set if a HFXT fault condition exists. HFXTOFFG can be cleared through software. If the HFXT fault condition still remains, HFXTOFFG is set. 0b = No fault condition occurred after the last reset 1b = HFXT fault; an HFXT fault occurred after the last reset
0	LFXTOFFG	RW	1h	LFXT oscillator fault flag. If this bit is set, the OFIFG flag is also set. LFXTOFFG is set if a LFXT fault condition exists. LFXTOFFG can be cleared through software. If the LFXT fault condition still remains, LFXTOFFG is set. 0b = No fault condition occurred after the last reset 1b = LFXT fault; an LFXT fault occurred after the last reset

Table 1-14. SFRIFG1 Register Description

Bit	Field	Type	Reset	Description
15-8	Reserved	R	0h	Reserved. Always reads as 0.
7	JMBOUTIFG	RW	1h	JTAG mailbox output interrupt flag 0b = No interrupt pending. When in 16-bit mode (JMBMODE = 0), this bit is cleared automatically when JMBO0 has been written with a new message to the JTAG module by the CPU. When in 32-bit mode (JMBMODE = 1), this bit is cleared automatically when both JMBO0 and JMBO1 have been written with new messages to the JTAG module by the CPU. This bit is also cleared when the associated vector in SYSUNIV has been read. 1b = Interrupt pending, JMBO registers are ready for new messages. In 16-bit mode (JMBMODE = 0), JMBO0 has been received by the JTAG module and is ready for a new message from the CPU. In 32-bit mode (JMBMODE = 1) , JMBO0 and JMBO1 have been received by the JTAG module and are ready for new messages from the CPU.
6	JMBINIFG	RW	0h	JTAG mailbox input interrupt flag 0b = No interrupt pending. When in 16-bit mode (JMBMODE = 0), this bit is cleared automatically when JMBI0 is read by the CPU. When in 32-bit mode (JMBMODE = 1), this bit is cleared automatically when both JMBI0 and JMBI1 have been read by the CPU. This bit is also cleared when the associated vector in SYSUNIV has been read 1b = Interrupt pending, a message is waiting in the JMBIN registers. In 16-bit mode (JMBMODE = 0) when JMBI0 has been written by the JTAG module. In 32-bit mode (JMBMODE = 1) when JMBI0 and JMBI1 have been written by the JTAG module.
5	Reserved	R	0h	Reserved. Always reads as 0.
4	NMIIFG	RW	0h	NMI pin interrupt flag 0b = No interrupt pending 1b = Interrupt pending
3	VMAIFG	RW	0h	Vacant memory access interrupt flag 0b = No interrupt pending 1b = Interrupt pending
2	Reserved	R	0h	Reserved. Always reads as 0.
1	OFIFG	RW	1h	Oscillator fault interrupt flag 0b = No interrupt pending 1b = Interrupt pending
0	WDTIFG	RW	0h	Watchdog timer interrupt flag. In watchdog mode, WDTIFG clears itself upon a watchdog timeout event. The SYSRSTIV can be read to determine if the reset was caused by a watchdog timeout event. In interval mode, WDTIFG is reset automatically by servicing the interrupt, or can be reset by software. Because other bits in SFRIFG1 may be used for other modules, it is recommended to set or clear WDTIFG by using BIS.B or BIC.B instructions, rather than MOV.B or CLR.B instructions. 0b = No interrupt pending 1b = Interrupt pending

```
do
{
  CSCTL5 &= ~LFXTOFFG;                  // Clear XT1 fault flag
  SFRIFG1 &= ~OFIFG;
}while (SFRIFG1&OFIFG);                 // Test oscillator fault flag
```

（7）晶振正常后所处时钟配置寄存器。

```
CSCTL0_H = 0;                           // Lock CS registers
```

6、配置USCI_A0为UART模式。

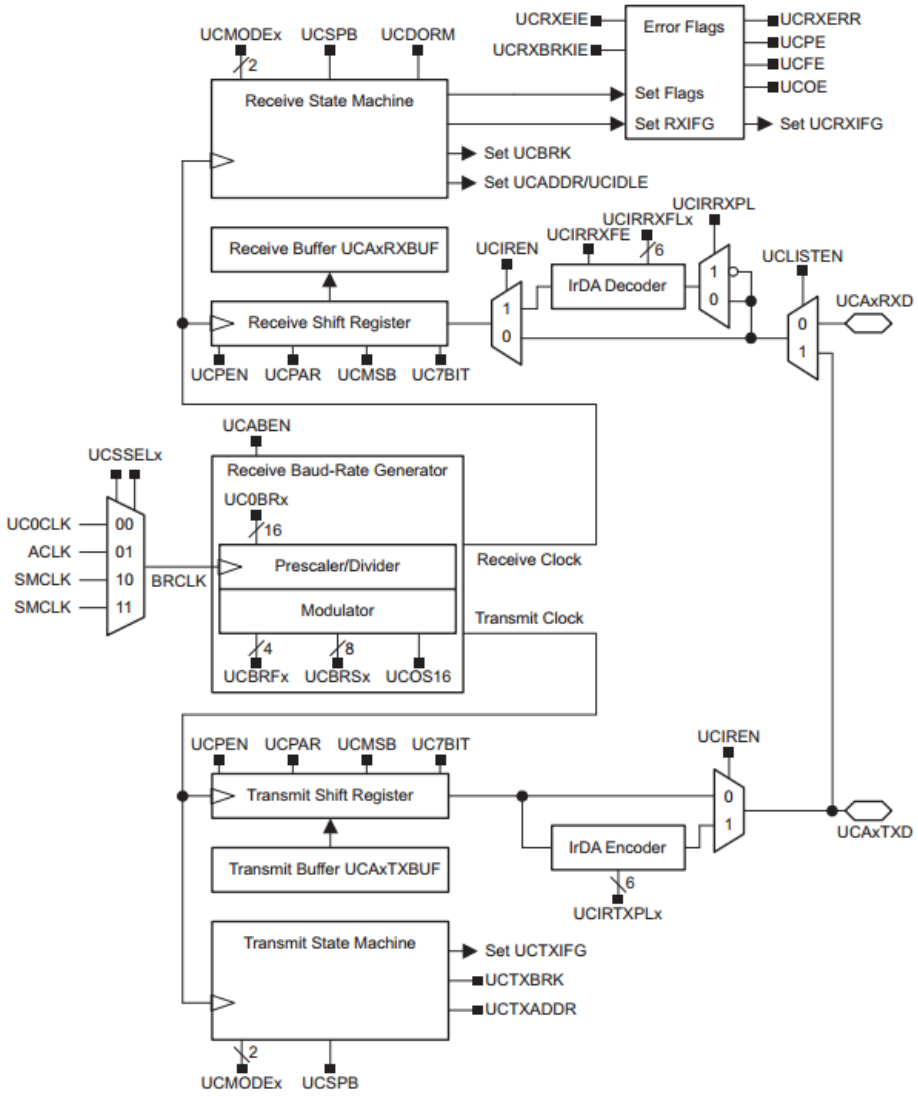


Figure 24-1. eUSCI_Ax Block Diagram – UART Mode (UCSYNC = 0)

（1）在配置或者重新配置eUSCI_A 模块时应该先将UCSWRST置1以避免出现位置错误，配置完之后释放复位。

```
UCA0CTLW0 = UCSWRST;                    // Put eUSCI in reset

UCA0CTL1 &= ~UCSWRST;                   // release from reset
```

24.4.1 UCAxCTLW0 Register

eUSCI_Ax Control Word Register 0

Figure 24-12. UCAxCTLW0 Register

15	14	13	12	11	10	9	8
UCPEN	UCPAR	UCMSB	UC7BIT	UCSPB	UCMODEx		UCSYNC
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
7	6	5	4	3	2	1	0
UCSSELx		UCRXEIE	UCBRKIE	UCDORM	UCTXADDR	UCTXBRK	UCSWRST
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-1

Can be modified only when UCSWRST = 1.

Table 24-8. UCAxCTLW0 Register Description

Bit	Field	Type	Reset	Description
15	UCPEN	RW	0h	Parity enable 0b = Parity disabled 1b = Parity enabled. Parity bit is generated (UCAxTXD) and expected (UCAxRXD). In address-bit multiprocessor mode, the address bit is included in the parity calculation.
14	UCPAR	RW	0h	Parity select. UCPAR is not used when parity is disabled. 0b = Odd parity 1b = Even parity
13	UCMSB	RW	0h	MSB first select. Controls the direction of the receive and transmit shift register. 0b = LSB first 1b = MSB first
12	UC7BIT	RW	0h	Character length. Selects 7-bit or 8-bit character length. 0b = 8-bit data 1b = 7-bit data
11	UCSPB	RW	0h	Stop bit select. Number of stop bits. 0b = One stop bit 1b = Two stop bits
10-9	UCMODEx	RW	0h	eUSCI_A mode. The UCMODEx bits select the asynchronous mode when UCSYNC = 0. 00b = UART mode 01b = Idle-line multiprocessor mode 10b = Address-bit multiprocessor mode 11b = UART mode with automatic baud-rate detection
8	UCSYNC	RW	0h	Synchronous mode enable 0b = Asynchronous mode 1b = Synchronous mode

7-6	UCSSELx	RW	0h	eUSCI_A clock source select. These bits select the BRCLK source clock. 00b = UCLK 01b = ACLK 10b = SMCLK 11b = SMCLK
5	UCRXEIE	RW	0h	Receive erroneous-character interrupt enable 0b = Erroneous characters rejected and UCRXIFG is not set. 1b = Erroneous characters received set UCRXIFG.
4	UCBRKIE	RW	0h	Receive break character interrupt enable 0b = Received break characters do not set UCRXIFG. 1b = Received break characters set UCRXIFG.

Table 24-8. UCAxCTLW0 Register Description (continued)

Bit	Field	Type	Reset	Description
3	UCDORM	RW	0h	Dormant. Puts eUSCI_A into sleep mode. 0b = Not dormant. All received characters set UCRXIFG. 1b = Dormant. Only characters that are preceded by an idle-line or with address bit set UCRXIFG. In UART mode with automatic baud-rate detection, only the combination of a break and synch field sets UCRXIFG.
2	UCTXADDR	RW	0h	Transmit address. Next frame to be transmitted is marked as address, depending on the selected multiprocessor mode. 0b = Next frame transmitted is data. 1b = Next frame transmitted is an address.
1	UCTXBRK	RW	0h	Transmit break. Transmits a break with the next write to the transmit buffer. In UART mode with automatic baud-rate detection, 055h must be written into UCAxTXBUF to generate the required break/synch fields. Otherwise, 0h must be written into the transmit buffer. 0b = Next frame transmitted is not a break. 1b = Next frame transmitted is a break or a break/synch.
0	UCSWRST	RW	1h	Software reset enable 0b = Disabled. eUSCI_A reset released for operation. 1b = Enabled. eUSCI_A logic held in reset state.

（2）设置eUSCI_A时钟源，作为波特率时钟源。

```
UCA0CTL1 |= UCSSEL__SMCLK;              // CLK = SMCLK
```

选择SMCLK为时钟源。

（3）设置波特率。

首先计算波特率参数。

对于给定的BRCLK时钟源，使用的波特率决定了所需的分频因子N：

N = f_{BRCLK} /波特率

分频因子N通常是非整数值，因此，使用至少一个分频器和一个调制器级来尽可能接近该因子。

如果N等于或大于16，TI建议通过设置UCOS16来使用过采样波特率生成模式。

注意：波特率设置快速设置

要计算波特率生成的正确设置，请执行以下步骤：

1.计算N = fBRCLK /波特率[如果N> 16继续步骤3，否则步骤2]

2. OS16 = 0，UCBRx = INT（N）[继续步骤4]

3. OS16 = 1，UCBRx = INT（N / 16），UCBRFx = INT（[（N / 16） - INT（N / 16）]×16）

可以通过查看表24-4中的N（= N-INT（N））的小数部分来找到UCBRSx

5.如果选择OS16 = 0，TI建议执行详细的错误计算。

Table 24-4. UCBRSx Settings for Fractional Portion of N = f_{BRCLK}/Baud Rate

Fractional Portion of N	UCBRSx[1]		Fractional Portion of N	UCBRSx[1]
0.0000	0x00		0.5002	0xAA
0.0529	0x01		0.5715	0x6B
0.0715	0x02		0.6003	0xAD
0.0835	0x04		0.6254	0xB5
0.1001	0x08		0.6432	0xB6
0.1252	0x10		0.6667	0xD6
0.1430	0x20		0.7001	0xB7
0.1670	0x11		0.7147	0xBB
0.2147	0x21		0.7503	0xDD
0.2224	0x22		0.7861	0xED
0.2503	0x44		0.8004	0xEE
0.3000	0x25		0.8333	0xBF
0.3335	0x49		0.8464	0xDF
0.3575	0x4A		0.8572	0xEF
0.3753	0x52		0.8751	0xF7
0.4003	0x92		0.9004	0xFB
0.4286	0x53		0.9170	0xFD
0.4378	0x55		0.9288	0xFE

(1) The UCBRSx setting in one row is valid from the fractional portion given in that row until the one in the next row

也可以直接查表Table 24-5确认波特率参数。

Table 24-5. Recommended Settings for Typical Crystals and Baud Rates[1]

BRCLK	Baud Rate	UCOS16	UCBRx	UCBRFx	UCBRSx[2]	TX Error[2] (%)		RX Error[2] (%)	
						neg	pos	neg	pos
32768	1200	1	1	11	0x25	-2.29	2.25	-2.56	5.35
32768	2400	0	13	-	0xB6	-3.12	3.91	-5.52	8.84
32768	4800	0	6	-	0xEE	-7.62	8.98	-21	10.25
32768	9600	0	3	-	0x92	-17.19	16.02	-23.24	37.3
1000000	9600	1	6	8	0x20	-0.48	0.64	-1.04	1.04
1000000	19200	1	3	4	0x2	-0.8	0.96	-1.84	1.84
1000000	38400	1	1	10	0x0	0	1.76	0	3.44
1000000	57600	0	17	-	0x4A	-2.72	2.56	-3.76	7.28
1000000	115200	0	8	-	0xD6	-7.36	5.6	-17.04	6.96
1048576	9600	1	6	13	0x22	-0.46	0.42	-0.48	1.23
1048576	19200	1	3	6	0xAD	-0.88	0.83	-2.36	1.18
1048576	38400	1	1	11	0x25	-2.29	2.25	-2.56	5.35
1048576	57600	0	18	-	0x11	-2	3.37	-5.31	5.55
1048576	115200	0	9	-	0x08	-5.37	4.49	-5.93	14.92
4000000	9600	1	26	0	0xB6	-0.08	0.16	-0.28	0.2
4000000	19200	1	13	0	0x84	-0.32	0.32	-0.64	0.48
4000000	38400	1	6	8	0x20	-0.48	0.64	-1.04	1.04
4000000	57600	1	4	5	0x55	-0.8	0.64	-1.12	1.76
4000000	115200	1	2	2	0xBB	-1.44	1.28	-3.92	1.68
4000000	230400	0	17	-	0x4A	-2.72	2.56	-3.76	7.28
4194304	9600	1	27	4	0xFB	-0.11	0.1	-0.33	0
4194304	19200	1	13	10	0x55	-0.21	0.21	-0.55	0.33
4194304	38400	1	6	13	0x22	-0.46	0.42	-0.48	1.23
4194304	57600	1	4	8	0xEE	-0.75	0.74	-2	0.87
4194304	115200	1	2	4	0x92	-1.62	1.37	-3.56	2.06
4194304	230400	0	18	-	0x11	-2	3.37	-5.31	5.55
8000000	9600	1	52	1	0x49	-0.08	0.04	-0.1	0.14
8000000	19200	1	26	0	0xB6	-0.08	0.16	-0.28	0.2
8000000	38400	1	13	0	0x84	-0.32	0.32	-0.64	0.48
8000000	57600	1	8	10	0xF7	-0.32	0.32	-1	0.36
8000000	115200	1	4	5	0x55	-0.8	0.64	-1.12	1.76
8000000	230400	1	2	2	0xBB	-1.44	1.28	-3.92	1.68
8000000	460800	0	17	-	0x4A	-2.72	2.56	-3.76	7.28
8388608	9600	1	54	9	0xEE	-0.06	0.06	-0.11	0.13
8388608	19200	1	27	4	0xFB	-0.11	0.1	-0.33	0
8388608	38400	1	13	10	0x55	-0.21	0.21	-0.55	0.33
8388608	57600	1	9	1	0xB5	-0.31	0.31	-0.53	0.78
8388608	115200	1	4	8	0xEE	-0.75	0.74	-2	0.87

Table 24-5. Recommended Settings for Typical Crystals and Baud Rates[1] (continued)

BRCLK	Baud Rate	UCOS16	UCBRx	UCBRFx	UCBRSx[2]	TX Error[2] (%)		RX Error[2] (%)	
						neg	pos	neg	pos
8388608	230400	1	2	4	0x92	-1.62	1.37	-3.56	2.06
8388608	460800	0	18	-	0x11	-2	3.37	-5.31	5.55
12000000	9600	1	78	2	0x0	0	0	0	0.04
12000000	19200	1	39	1	0x0	0	0	0	0.16
12000000	38400	1	19	8	0x65	-0.16	0.16	-0.4	0.24
12000000	57600	1	13	0	0x25	-0.16	0.32	-0.48	0.48
12000000	115200	1	6	8	0x20	-0.48	0.64	-1.04	1.04
12000000	230400	1	3	4	0x2	-0.8	0.96	-1.84	1.84
12000000	460800	1	1	10	0x0	0	1.76	0	3.44
16000000	9600	1	104	2	0xD6	-0.04	0.02	-0.09	0.03
16000000	19200	1	52	1	0x49	-0.08	0.04	-0.1	0.14
16000000	38400	1	26	0	0xB6	-0.08	0.16	-0.28	0.2
16000000	57600	1	17	5	0xDD	-0.16	0.2	-0.3	0.38
16000000	115200	1	8	10	0xF7	-0.32	0.32	-1	0.36
16000000	230400	1	4	5	0x55	-0.8	0.64	-1.12	1.76
16000000	460800	1	2	2	0xBB	-1.44	1.28	-3.92	1.68
16777216	9600	1	109	3	0xB5	-0.03	0.02	-0.05	0.06
16777216	19200	1	54	9	0xEE	-0.06	0.06	-0.11	0.13
16777216	38400	1	27	4	0xFB	-0.11	0.1	-0.33	0
16777216	57600	1	18	3	0x44	-0.16	0.15	-0.2	0.45
16777216	115200	1	9	1	0xB5	-0.31	0.31	-0.53	0.78
16777216	230400	1	4	8	0xEE	-0.75	0.74	-2	0.87
16777216	460800	1	2	4	0x92	-1.62	1.37	-3.56	2.06
20000000	9600	1	130	3	0x25	-0.02	0.03	0	0.07
20000000	19200	1	65	1	0xD6	-0.06	0.03	-0.1	0.1
20000000	38400	1	32	8	0xEE	-0.1	0.13	-0.27	0.14
20000000	57600	1	21	11	0x22	-0.16	0.13	-0.16	0.38
20000000	115200	1	10	13	0xAD	-0.29	0.26	-0.46	0.66
20000000	230400	1	5	6	0xEE	-0.67	0.51	-1.71	0.62
20000000	460800	1	2	11	0x92	-1.38	0.99	-1.84	2.8

24.4.3 UCAxBRW Register

eUSCI_Ax Baud Rate Control Word Register

Figure 24-14. UCAxBRW Register

15	14	13	12	11	10	9	8
UCBRx							
rw	rw	rw	rw	rw	rw	rw	rw
7	6	5	4	3	2	1	0
UCBRx							
rw	rw	rw	rw	rw	rw	rw	rw

Can be modified only when UCSWRST = 1.

Table 24-10. UCAxBRW Register Description

Bit	Field	Type	Reset	Description
15-0	UCBRx	RW	0h	Clock prescaler setting of the Baud rate generator

24.4.4 UCAxMCTLW Register

eUSCI_Ax Modulation Control Word Register

Figure 24-15. UCAxMCTLW Register

15	14	13	12	11	10	9	8
UCBRSx							
rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0	rw-0
7	6	5	4	3	2	1	0
UCBRFx				Reserved			UCOS16
rw-0	rw-0	rw-0	rw-0	r0	r0	r0	rw-0

Can be modified only when UCSWRST = 1.

Table 24-11. UCAxMCTLW Register Description

Bit	Field	Type	Reset	Description
15-8	UCBRSx	RW	0h	Second modulation stage select. These bits hold a free modulation pattern for BITCLK.
7-4	UCBRFx	RW	0h	First modulation stage select. These bits determine the modulation pattern for BITCLK16 when UCOS16 = 1. Ignored with UCOS16 = 0. The "Oversampling Baud-Rate Generation" section shows the modulation pattern.
3-1	Reserved	R	0h	Reserved
0	UCOS16	RW	0h	Oversampling mode enabled 0b = Disabled 1b = Enabled

UCBRx = INT(N)

```
UCA0BR0 = 8;                    // 1000000/115200 = 8.68
```

因为N<16,所以OS16=0，UCBRFx不需要写， 由8.68-8=0.68查Table 24-4得UCBRSx=0xD6

```
UCA0MCTLW = 0xD600;                     // 1000000/115200 - INT(1000000/115200)=0.68
                                        // UCBRSx value = 0xD6 (See UG)
UCA0BR1 = 0;
```

接下来使能中断

```
UCA0IE |= UCRXIE;                       // Enable USCI_A0 RX interrupt
```

24.4.10 UCAxIE Register

eUSCI_Ax Interrupt Enable Register

Figure 24-21. UCAxIE Register

15	14	13	12	11	10	9	8
Reserved							
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0
7	6	5	4	3	2	1	0
Reserved				UCTXCPTIE	UCSTTIE	UCTXIE	UCRXIE
r-0	r-0	r-0	r-0	rw-0	rw-0	rw-0	rw-0

Table 24-17. UCAxIE Register Description

Bit	Field	Type	Reset	Description
15-4	Reserved	R	0h	Reserved
3	UCTXCPTIE	RW	0h	Transmit complete interrupt enable 0b = Interrupt disabled 1b = Interrupt enabled
2	UCSTTIE	RW	0h	Start bit interrupt enable 0b = Interrupt disabled 1b = Interrupt enabled
1	UCTXIE	RW	0h	Transmit interrupt enable 0b = Interrupt disabled 1b = Interrupt enabled
0	UCRXIE	RW	0h	Receive interrupt enable 0b = Interrupt disabled 1b = Interrupt enabled

7、通过中断标志判断状态

24.4.11 UCAxIFG Register

eUSCI_Ax Interrupt Flag Register

Figure 24-22. UCAxIFG Register

15	14	13	12	11	10	9	8
Reserved							
r-0	r-0	r-0	r-0	r-0	r-0	r-0	r-0
7	6	5	4	3	2	1	0
Reserved				UCTXCPTIFG	UCSTTIFG	UCTXIFG	UCRXIFG
r-0	r-0	r-0	r-0	rw-0	rw-0	rw-1	rw-0

Table 24-18. UCAxIFG Register Description

Bit	Field	Type	Reset	Description
15-4	Reserved	R	0h	Reserved
3	UCTXCPTIFG	RW	0h	Transmit complete interrupt flag. UCTXCPTIFG is set when the entire byte in the internal shift register got shifted out and UCAxTXBUF is empty. 0b = No interrupt pending 1b = Interrupt pending
2	UCSTTIFG	RW	0h	Start bit interrupt flag. UCSTTIFG is set after a Start bit was received 0b = No interrupt pending 1b = Interrupt pending
1	UCTXIFG	RW	1h	Transmit interrupt flag. UCTXIFG is set when UCAxTXBUF empty. 0b = No interrupt pending 1b = Interrupt pending
0	UCRXIFG	RW	0h	Receive interrupt flag. UCRXIFG is set when UCAxRXBUF has received a complete character. 0b = No interrupt pending 1b = Interrupt pending

```
while (1)
{
  while(!(UCA0IFG & UCTXIFG));
  UCA0TXBUF = TXData++;                 // Load data onto buffer
}
```

当发送完成后往发送buffer装载数据。

8、中断函数子程序。

可以参考下面的例程：

Example 24-1. UCAxIV Software Example

```
#pragma vector = USCI_A0_VECTOR __interrupt void USCI_A0_ISR(void) {
  switch(__even_in_range(UCA0IV,18)) {
    case 0x00: // Vector 0: No interrupts
      break;
    case 0x02: ... // Vector 2: UCRXIFG
      break;
    case 0x04: ... // Vector 4: UCTXIFG
      break;
    case 0x06: ... // Vector 6: UCSTTIFG
```

```
        break;
      case 0x08: ... // Vector 8: UCTXCPTIFG
        break;
      default: break;
    }
}
```

其中，__even_in_range(UCA0IV,18)的解释：

使用__even_in_range 的好处是可以生成效率比较高的代码，在判断多中断源的中断的来源时可以使用此函数。

原型：unsigned short __even_in_range(unsigned short value, unsignedshort upper_limit);

功能：只能与switch 语句结合使用，判断value 是否为偶数且小于等于upper_limit。