

```

/*
-----
2.996 / 6.971: Biomedical Devices Design Laboratory
Lab Example Software - UART Communication

This example sets up the UART at 9600bps, no parity, one
stop bit (XBee default). It will echo back characters it
receives from the USB or XBee radio. Test with a terminal
program.

SC - 9/30/2007
-----
*/

#include "msp430x22x4.h"

// special interrupt routine syntax
#pragma vector=USCIAB0RX_VECTOR // defined in "msp430x22x4.h"
__interrupt void UART_RX(void) // interrupt on incoming data
{
    unsigned char rx_byte;
    rx_byte = UCA0RXBUF; // grab byte from RX buffer

    while((IFG2 & UCA0TXIFG) == 0); // wait for TX buffer to be clear
    UCA0TXBUF = rx_byte; // transmit the received data byte

    /*
    Note: There are two methods for ensuring cleanly-transmitted
    data. Waiting for the TX buffer to clear, as above, will lock
    up your program if you are transmitting several bytes as it waits
    for the buffer to clear each time. Use TX interrupts for
    efficient asynchronous transmission. See User's Guide, 15-25
    thru 15-26.
    */
}

void main(void)
{
    // stop watchdog timer
    WDTCTL = WDTPW | WDTHOLD;

    // Clock Setup:
    // -----
    // XT2 not used, LFXT1 set to high-frequency mode
    // no divider for ACLK (full 16MHz)
    BCSCTL1 = XT2OFF | XTS;
    // set MCLK as LFXT1 (16MHz), no divider
    // also set SMCLK as LFXT1, but divide by 4 (4MHz)
    BCSCTL2 = SELM1 | SELM0 | SELS | DIVS1;
    // set LFXT1 to 3-16MHz range
    BCSCTL3 = LFXT1S1;
    // See User's Guide, 5-14 thru 5-16.
    // -----

    // Pin Setup:
    // -----
    // set P3.4 (TX) as output, P3.5 (RX) is input by default
    P3DIR = BIT4;
    // select P3.4 and P3.5 to be controlled by the UART
    P3SEL = BIT4 | BIT5;
    // -----

    // UART Setup:
    // -----
    // use ACLK (16MHz) as baud rate clock
    UCA0CTL1 = UCSSEL0;
    // UCA0BR0 = int(fCLK/baud rate)
    // = int(16000000/9600) = 1666
    // divide into low and high byte
    UCA0BR0 = 1666 & 0xFF; UCA0BR1 = 1666 >> 8;
    // modulation, not very important at low baud rates

```

```
UCA0MCTL = UCBRF_0 | UCBSR_6;
// enable receive data interrupt
IE2 |= UCA0RXIE;
// See User's Guide, 15-28 thru 15-35
// and 15-22 thru 15-23 for common baud rate settings
// -----

while(1); // loop forever

}
```