```
32        // Startup clock system with max DCO setting ~8MHz
33        CSCTL0_H = CSKEY >> 8;                     // Unlock clock registers (contrôle d'accées aux résistres)
34        CSCTL1 = DCOFSEL_3 | DCORSEL;             // Set DCO to 8MHz
35        CSCTL2 = SELA__VLOCLK | SELS__DCOCLK | SELM__DCOCLK;
36        CSCTL3 = DIVA__1 | DIVS__1 | DIVM__1;     // Set all dividers
37        CSCTL0_H = 0;                             // Lock CS registers
38
39        // Configure USCI_A0 for UART mode
40        UCA0CTLW0 = UCSWRST;                      // Put eUSCI in reset
41        UCA0CTLW0 |= UCSSEL__SMCLK;               // CLK = SMCLK
42        // Baud Rate calculation
43        // 8000000/(16*9600) = 52.083
44        // Fractional portion = 0.083
45        // User's Guide Table 21-4: UCBRSx = 0x04
46        // UCBRFx = int ( (52.083-52)*16) = 1
47        UCA0BR0 = 52;                             // 8000000/16/9600
48        UCA0BR1 = 0x00;
49        UCA0MCTLW |= UCOS16 | UCBRF_1;
50        UCA0CTLW0 &= ~UCSWRST;                    // Initialize eUSCI
51        UCA0IE |= UCRXIE;                         // Enable USCI_A0 RX interrupt
52
53        __bis_SR_register(LPM3_bits | GIE);       // Enter LPM3, interrupts enabled
54        __no_operation();                         // For debugger
55    //}
56
57    /* #if defined(__TI_COMPILER_VERSION__) || defined(__IAR_SYSTEMS_ICC__)
58     #pragma vector=USCI_A0_VECTOR
59     __interrupt void USCI_A0_ISR(void)
60     #elif defined(__GNUC__)
61     void __attribute__ ((interrupt(USCI_A0_VECTOR))) USCI_A0_ISR (void)
62     #else
63     #error Compiler not supported!
64     #endif
65     {
66       switch(__even_in_range(UCA0IV, USCI_UART_UCTXCPTIFG))
67       {
68         case USCI_NONE: break;
69         case USCI_UART_UCRXIFG:
70           while(!(UCA0IFG&UCTXIFG));
71           UCA0TXBUF = UCA0RXBUF;
72           __no_operation();
73           break;
74         case USCI_UART_UCTXIFG: break;
75         case USCI_UART_UCSTTIFG: break;
76         case USCI_UART_UCTXCPTIFG: break;
77       } */
78
```