```
 1: /*
 2:   Initiating ADC Conversion:
 3:     1. Writing to ATD0CTL5
 4:     2. Edge on external trigger
 5:     3. Level on external trigger
 6:
 7:   Completed Conversion
 8:     1. Reading ATD0STAT1
 9:     2. Interrupt when complete
10: */
11:
12: #include <hidef.h>      /* common defines and macros */
13: #include <mc9s12dp512.h>     /* derivative information */
14: #pragma LINK_INFO DERIVATIVE "mc9s12dp512"
15:
16: #include "PLL.h"
17: #include "lcd.h"
18: #include "Timer.h"
19: #include <stdio.h>
20:
21: #include "SCI1.h"
22: #include "Fifo.h"
23: #include "Xbee.h"
24:
25: volatile char whee;
26:
27: void main(void) {
28:   int i;
29:   DDRP |= 0x80;
30:   PLL_Init(); // 24 MHz
31:   Timer_Init();
32:   LCD_Open();
33:   Fifo_Init();
34:   SCI1_Init(9600); // SCI output to PC
35:
36:   asm cli;
37:
38:     LCD_Clear();
39:
40:   XBee_Init();
41:   Timer_Wait10ms(100);
42:   for(;;) {
43:     long blah;
44:     FrameType frame;
45:     Fifo_Init();
46:     while(!XBee_RecieveTxFrame(&frame));
47:
48:     LCD_Clear();
49:     for(i=5; i<frame.length; i++) {
50:       if(i == 13) {
51:         LCD_GoTo(1,0);
52:       }
53:       LCD_OutChar(frame.data[i]);
54:     }
55:     Fifo_Init();
56:     PTP ^= 0x80;
57:   }
58: }
```

```
  1: #include "derivative.h"      /* derivative-specific definitions */
  2: #include "SCI1.h"
  3: #include "Timer.h"
  4: #include "XBee.h"
  5:
  6: int ok(void) {
  7:   if(SCI1_InChar() != 'O') {
  8:     return 0;
  9:   }
 10:   if(SCI1_InChar() != 'K') {
 11:     return 0;
 12:   }
 13:   if(SCI1_InChar() != 0x0D) {
 14:     return 0;
 15:   }
 16:
 17:   return 1;
 18: }
 19:
 20:
 21: void sendATCommand(char * command) {
 22:   char * temp;
 23:   do {
 24:     temp = command;
 25:     while(*temp) {
 26:       SCI1_OutChar(*temp);
 27:       temp++;
 28:     }
 29:     SCI1_OutChar(0x0D);
 30:     Timer_Wait1ms(20);
 31:   } while(!ok());
 32: }
 33:
 34: /*----------------------XBee_Init----------------------------
 35:   Initialize XBee
 36:   Inputs: none
 37:   Outputs: none */
 38: void XBee_Init(void) {
 39:     SCI1_OutChar('X');
 40:     Timer_Wait10ms(110);
 41:
 42:     SCI1_OutString("+++");
 43:     Timer_Wait10ms(110);
 44:
 45:     sendATCommand("ATDL4F");
 46:     sendATCommand("ATDH0");
 47:     sendATCommand("ATMY4E");
 48:     sendATCommand("ATAP1");
 49:     sendATCommand("ATCN");
 50:     PTP ^= 0x80;
 51: }
 52:
 53: /*----------------------XBee_RecieveTxFrame------------------
 54:   Receives a frame from data in
 55:   Inputs: None
 56:   Outputs: Input Frame */
 57: int XBee_RecieveTxFrame(FrameType * frame)
 58: {
 59:   short i;
 60:   static short FrameID = 1;
 61:
 62:   if(SCI1_InChar() != 0x7E)
 63:   {
 64:     return 0;
 65:   }
 66:
 67:   frame->length = SCI1_InChar();
 68:   frame->length <<= 8;
 69:   frame->length += SCI1_InChar();
 70:
 71:   for(i = 0; i < frame->length; i++)
 72:   {
 73:     frame->data[i] = SCI1_InChar();
 74:   }
 75:
 76:   frame->checkSum = SCI1_InChar();
 77:   frame->frameID = FrameID++;
 78:   return 1;
```

```
 79: }
```

```
 1: typedef struct
 2: {
 3:   short length; // Length of data
 4:   char data[30];
 5:   char checkSum;
 6:   short frameID;
 7: } FrameType;
 8:
 9:
10:
11: /*-----------------------XBee_Init----------------------------
12:   Initialize XBee
13:   Inputs: none
14:   Outputs: none */
15: void XBee_Init(void);
16:
17: /*-----------------------XBee_RecieveTxFrame-------------------
18:   Receives a frame from data in
19:   Inputs: None
20:   Outputs: Input Frame */
21: int XBee_RecieveTxFrame(FrameType * frame);
```