```c
#include "globals.h"

int main(int argc, char *argv[]) {
  xbee_con *con, *con2;
  xbee_pkt *pkt;
  int i;

  xbee_setup("/dev/ttyUSB1",57600);

  /*if ((con = xbee_newcon(NULL,'X',xbee_localAT)) == (void *)-1) {
    printf("error creating connection...\n");
    exit(1);
  }

  while(1){sleep(10);}

  xbee_senddata(con,"CH%c",0x0C);
  sleep(1);
  xbee_senddata(con,"ID%c%c",0x33, 0x32);
  sleep(1);
  xbee_senddata(con,"DH%c%c%c%c",0x00,0x00,0x00,0x00);
  sleep(1);
  xbee_senddata(con,"DL%c%c%c%c",0x00,0x00,0x00,0x00);
  sleep(1);
  xbee_senddata(con,"MY%c%c",0x00,0x00);
  sleep(1);
  // SH - read only
  // SL - read only
  xbee_senddata(con,"RR%c",0x00);
  sleep(1);
  xbee_senddata(con,"RN%c",0x00);
  sleep(1);
  xbee_senddata(con,"MM%c",0x00);
  sleep(1);
  xbee_senddata(con,"NT%c",0x19);
  sleep(1);
  xbee_senddata(con,"NO%c",0x00);
  sleep(1);
  xbee_senddata(con,"CE%c",0x00);
  sleep(1);
  xbee_senddata(con,"SC%c%c",0x1F,0xFE);
  sleep(1);
  xbee_senddata(con,"SD%c",0x04);
  sleep(1);
  xbee_senddata(con,"A1%c",0x00);
  sleep(1);
  xbee_senddata(con,"A2%c",0x00);
  sleep(1);
  // AI - read only
  xbee_senddata(con,"EE%c",0x00);
  sleep(1);
  //xbee_senddata(con,"KY%c",0x00);
  //sleep(1);
  xbee_senddata(con,"NI%s","TIGGER");
  sleep(1);
  xbee_senddata(con,"PL%c",0x04);
  sleep(1);
  xbee_senddata(con,"CA%c",0x2C);
  sleep(1);
  xbee_senddata(con,"SM%c",0x00);
  sleep(1);
  xbee_senddata(con,"ST%c%c",0x13,0x88);
  sleep(1);
  xbee_senddata(con,"SP%c%c",0x00,0x00);
  sleep(1);
  xbee_senddata(con,"DP%c%c",0x03,0xE8);
  sleep(1);
  xbee_senddata(con,"SO%c",0x00);
  sleep(1);
  xbee_senddata(con,"BD%c",0x06);
  sleep(1);
  xbee_senddata(con,"RO%c",0x03);
  sleep(1);
  xbee_senddata(con,"AP%c",0x02);
  sleep(1);
  xbee_senddata(con,"PR%c",0xFF);
  sleep(1);
  xbee_senddata(con,"D8%c",0x00);
  sleep(1);
  xbee_senddata(con,"D7%c",0x01);
  sleep(1);
  xbee_senddata(con,"D6%c",0x00);
  sleep(1);
  xbee_senddata(con,"D5%c",0x01);
  sleep(1);
```

```
  xbee_senddata(con,"D4%c",0x00);
  sleep(1);
  xbee_senddata(con,"D3%c",0x00);
  sleep(1);
  xbee_senddata(con,"D2%c",0x00);
  sleep(1);
  xbee_senddata(con,"D1%c",0x00);
  sleep(1);
  xbee_senddata(con,"D0%c",0x00);
  sleep(1);
  xbee_senddata(con,"IU%c",0x00);
  sleep(1);
  xbee_senddata(con,"IT%c",0x01);
  sleep(1);
  xbee_senddata(con,"IC%c",0x00);
  sleep(1);
  xbee_senddata(con,"IR%c%c",0x00,0x00);
  sleep(1);
  xbee_senddata(con,"IA%c%c%c%c%c%c%c%c",0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF);
  sleep(1);
  xbee_senddata(con,"T0%c",0xFF);
  sleep(1);
  xbee_senddata(con,"T1%c",0xFF);
  sleep(1);
  xbee_senddata(con,"T2%c",0xFF);
  sleep(1);
  xbee_senddata(con,"T3%c",0xFF);
  sleep(1);
  xbee_senddata(con,"T4%c",0xFF);
  sleep(1);
  xbee_senddata(con,"T5%c",0xFF);
  sleep(1);
  xbee_senddata(con,"T6%c",0xFF);
  sleep(1);
  xbee_senddata(con,"T7%c",0xFF);
  sleep(1);
  xbee_senddata(con,"P0%c",0x01);
  sleep(1);
  xbee_senddata(con,"P1%c",0x00);
  sleep(1);
  xbee_senddata(con,"PT%c",0xFF);
  sleep(1);
  xbee_senddata(con,"RP%c",0x28);
  sleep(1);
  // VR - read only
  // HV - read only
  // DB - read only
  // EC - read only
  // EA - read only
  // DD - read only
  xbee_senddata(con,"CT%c",0x64);
  sleep(1);
  xbee_senddata(con,"GT%c%c",0x03,0xE8);
  sleep(1);
  xbee_senddata(con,"CC%c",0x2B);
  sleep(1);

  sleep(10);
  */

  con =  xbee_newcon('I',xbee_64bitIO,   0x0013A200, 0x40081826);
  con2 = xbee_newcon('I',xbee_64bitData, 0x0013A200, 0x40081826);

  while (1) {
    while ((pkt = xbee_getpacket(con)) != NULL) {
      printf("--------- got one!... CON2 -----------\n");
      if (pkt->IOmask & 0x0001) printf("Digital 0: %c\n",((pkt->IOdata & 0x0001)?'1':'0'));
      if (pkt->IOmask & 0x0002) printf("Digital 1: %c\n",((pkt->IOdata & 0x0002)?'1':'0'));
      if (pkt->IOmask & 0x0004) printf("Digital 2: %c\n",((pkt->IOdata & 0x0004)?'1':'0'));
      if (pkt->IOmask & 0x0008) printf("Digital 3: %c\n",((pkt->IOdata & 0x0008)?'1':'0'));
      if (pkt->IOmask & 0x0010) printf("Digital 4: %c\n",((pkt->IOdata & 0x0010)?'1':'0'));
      if (pkt->IOmask & 0x0020) printf("Digital 5: %c\n",((pkt->IOdata & 0x0020)?'1':'0'));
      if (pkt->IOmask & 0x0040) printf("Digital 6: %c\n",((pkt->IOdata & 0x0040)?'1':'0'));
      if (pkt->IOmask & 0x0080) printf("Digital 7: %c\n",((pkt->IOdata & 0x0080)?'1':'0'));
      if (pkt->IOmask & 0x0100) printf("Digital 8: %c\n",((pkt->IOdata & 0x0100)?'1':'0'));
      if (pkt->IOmask & 0x0200) printf("Analog  0: %.2fv\n",(3.3/1023)*pkt->IOanalog[0]);
      if (pkt->IOmask & 0x0400) printf("Analog  1: %.2fv\n",(3.3/1023)*pkt->IOanalog[1]);
      if (pkt->IOmask & 0x0800) printf("Analog  2: %.2fv\n",(3.3/1023)*pkt->IOanalog[2]);
      if (pkt->IOmask & 0x1000) printf("Analog  3: %.2fv\n",(3.3/1023)*pkt->IOanalog[3]);
      if (pkt->IOmask & 0x2000) printf("Analog  4: %.2fv\n",(3.3/1023)*pkt->IOanalog[4]);
      if (pkt->IOmask & 0x4000) printf("Analog  5: %.2fv\n",(3.3/1023)*pkt->IOanalog[5]);
      xbee_senddata(con2, "thank you %s %d\r", "so much", time(NULL));
      free(pkt);
    }
    usleep(100000);
```

```
171:   }
172:
173:   return 0;
174: }
```