

```
1: #include "globals.h"
2:
3: int main(int argc, char *argv[]) {
4:     xbee_con *con, *con2;
5:     xbee_pkt *pkt;
6:     int i;
7:
8:     xbee_setup("/dev/ttyUSB1", 57600);
9:
10:    /*if ((con = xbee_newcon(NULL, 'X', xbee_localAT)) == (void *)-1) {
11:        printf("error creating connection...\n");
12:        exit(1);
13:    }
14:
15:    while(1){sleep(10);}
16:
17:    xbee_senddata(con, "CH%c", 0x0C);
18:    sleep(1);
19:    xbee_senddata(con, "ID%c%c", 0x33, 0x32);
20:    sleep(1);
21:    xbee_senddata(con, "DH%c%c%c%c", 0x00, 0x00, 0x00, 0x00);
22:    sleep(1);
23:    xbee_senddata(con, "DL%c%c%c%c", 0x00, 0x00, 0x00, 0x00);
24:    sleep(1);
25:    xbee_senddata(con, "MY%c%c", 0x00, 0x00);
26:    sleep(1);
27:    // SH - read only
28:    // SL - read only
29:    xbee_senddata(con, "RR%c", 0x00);
30:    sleep(1);
31:    xbee_senddata(con, "RN%c", 0x00);
32:    sleep(1);
33:    xbee_senddata(con, "MM%c", 0x00);
34:    sleep(1);
35:    xbee_senddata(con, "NT%c", 0x19);
36:    sleep(1);
37:    xbee_senddata(con, "NO%c", 0x00);
38:    sleep(1);
39:    xbee_senddata(con, "CE%c", 0x00);
40:    sleep(1);
41:    xbee_senddata(con, "SC%c%c", 0x1F, 0xFE);
42:    sleep(1);
43:    xbee_senddata(con, "SD%c", 0x04);
44:    sleep(1);
45:    xbee_senddata(con, "A1%c", 0x00);
46:    sleep(1);
47:    xbee_senddata(con, "A2%c", 0x00);
48:    sleep(1);
49:    // AI - read only
50:    xbee_senddata(con, "EE%c", 0x00);
51:    sleep(1);
52:    //xbee_senddata(con, "KY%c", 0x00);
53:    //sleep(1);
54:    xbee_senddata(con, "NIS", "TIGGER");
55:    sleep(1);
56:    xbee_senddata(con, "PL%c", 0x04);
57:    sleep(1);
58:    xbee_senddata(con, "CA%c", 0x2C);
59:    sleep(1);
60:    xbee_senddata(con, "SM%c", 0x00);
61:    sleep(1);
62:    xbee_senddata(con, "ST%c%c", 0x13, 0x88);
63:    sleep(1);
64:    xbee_senddata(con, "SP%c%c", 0x00, 0x00);
65:    sleep(1);
66:    xbee_senddata(con, "DP%c%c", 0x03, 0xE8);
67:    sleep(1);
68:    xbee_senddata(con, "SO%c", 0x00);
69:    sleep(1);
70:    xbee_senddata(con, "BD%c", 0x06);
71:    sleep(1);
72:    xbee_senddata(con, "RO%c", 0x03);
73:    sleep(1);
74:    xbee_senddata(con, "AP%c", 0x02);
75:    sleep(1);
76:    xbee_senddata(con, "PR%c", 0xFF);
77:    sleep(1);
78:    xbee_senddata(con, "D8%c", 0x00);
79:    sleep(1);
80:    xbee_senddata(con, "D7%c", 0x01);
81:    sleep(1);
82:    xbee_senddata(con, "D6%c", 0x00);
83:    sleep(1);
84:    xbee_senddata(con, "D5%c", 0x01);
85:    sleep(1);
```

```

86:  xbee_senddata(con,"D4%c",0x00);
87:  sleep(1);
88:  xbee_senddata(con,"D3%c",0x00);
89:  sleep(1);
90:  xbee_senddata(con,"D2%c",0x00);
91:  sleep(1);
92:  xbee_senddata(con,"D1%c",0x00);
93:  sleep(1);
94:  xbee_senddata(con,"D0%c",0x00);
95:  sleep(1);
96:  xbee_senddata(con,"IU%c",0x00);
97:  sleep(1);
98:  xbee_senddata(con,"IT%c",0x01);
99:  sleep(1);
100: xbee_senddata(con,"IC%c",0x00);
101: sleep(1);
102: xbee_senddata(con,"IR%c%c",0x00,0x00);
103: sleep(1);
104: xbee_senddata(con,"IA%c%c%c%c%c%c%c",0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF);
105: sleep(1);
106: xbee_senddata(con,"T0%c",0xFF);
107: sleep(1);
108: xbee_senddata(con,"T1%c",0xFF);
109: sleep(1);
110: xbee_senddata(con,"T2%c",0xFF);
111: sleep(1);
112: xbee_senddata(con,"T3%c",0xFF);
113: sleep(1);
114: xbee_senddata(con,"T4%c",0xFF);
115: sleep(1);
116: xbee_senddata(con,"T5%c",0xFF);
117: sleep(1);
118: xbee_senddata(con,"T6%c",0xFF);
119: sleep(1);
120: xbee_senddata(con,"T7%c",0xFF);
121: sleep(1);
122: xbee_senddata(con,"P0%c",0x01);
123: sleep(1);
124: xbee_senddata(con,"P1%c",0x00);
125: sleep(1);
126: xbee_senddata(con,"PT%c",0xFF);
127: sleep(1);
128: xbee_senddata(con,"RP%c",0x28);
129: sleep(1);
130: // VR - read only
131: // HV - read only
132: // DB - read only
133: // EC - read only
134: // EA - read only
135: // DD - read only
136: xbee_senddata(con,"CT%c",0x64);
137: sleep(1);
138: xbee_senddata(con,"GT%c%c",0x03,0xE8);
139: sleep(1);
140: xbee_senddata(con,"CC%c",0x2B);
141: sleep(1);
142:
143: sleep(10);
144: */
145:
146: con = xbee_newcon('I',xbee_64bitIO, 0x0013A200, 0x40081826);
147: con2 = xbee_newcon('I',xbee_64bitData, 0x0013A200, 0x40081826);
148:
149: while (1) {
150:     while ((pkt = xbee_getpacket(con)) != NULL) {
151:         printf("----- got one!... CON2 ----- \n");
152:         if (pkt->Iomask & 0x0001) printf("Digital 0: %c\n",((pkt->Iodata & 0x0001)?'1':'0'));
153:         if (pkt->Iomask & 0x0002) printf("Digital 1: %c\n",((pkt->Iodata & 0x0002)?'1':'0'));
154:         if (pkt->Iomask & 0x0004) printf("Digital 2: %c\n",((pkt->Iodata & 0x0004)?'1':'0'));
155:         if (pkt->Iomask & 0x0008) printf("Digital 3: %c\n",((pkt->Iodata & 0x0008)?'1':'0'));
156:         if (pkt->Iomask & 0x0010) printf("Digital 4: %c\n",((pkt->Iodata & 0x0010)?'1':'0'));
157:         if (pkt->Iomask & 0x0020) printf("Digital 5: %c\n",((pkt->Iodata & 0x0020)?'1':'0'));
158:         if (pkt->Iomask & 0x0040) printf("Digital 6: %c\n",((pkt->Iodata & 0x0040)?'1':'0'));
159:         if (pkt->Iomask & 0x0080) printf("Digital 7: %c\n",((pkt->Iodata & 0x0080)?'1':'0'));
160:         if (pkt->Iomask & 0x0100) printf("Digital 8: %c\n",((pkt->Iodata & 0x0100)?'1':'0'));
161:         if (pkt->Iomask & 0x0200) printf("Analog 0: %.2fv\n", (3.3/1023)*pkt->IOanalog[0]);
162:         if (pkt->Iomask & 0x0400) printf("Analog 1: %.2fv\n", (3.3/1023)*pkt->IOanalog[1]);
163:         if (pkt->Iomask & 0x0800) printf("Analog 2: %.2fv\n", (3.3/1023)*pkt->IOanalog[2]);
164:         if (pkt->Iomask & 0x1000) printf("Analog 3: %.2fv\n", (3.3/1023)*pkt->IOanalog[3]);
165:         if (pkt->Iomask & 0x2000) printf("Analog 4: %.2fv\n", (3.3/1023)*pkt->IOanalog[4]);
166:         if (pkt->Iomask & 0x4000) printf("Analog 5: %.2fv\n", (3.3/1023)*pkt->IOanalog[5]);
167:         xbee_senddata(con2, "thank you %s %d\r", "so much", time(NULL));
168:         free(pkt);
169:     }
170:     usleep(100000);

```

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