

```
1: /*
2: Program to interface between N1MM Plus Logger and a transverter bandswitch.
3: */
4:
5: //include string handling library
6: #include <string.h>
7:
8: //define variables
9: String commandInputString = ""; // input buffer string to hold incoming
    data
10: boolean commandStringComplete = false; // true when the input string is complete
11: String command = ""; // incoming data string for parsing
12:
13:
14:
15: //define constant pin aliases
16: const int Pin50 = 2; //number of 50 MHz pin
17: const int Pin144 = 3; //number of 144 MHz pin
18: const int Pin222 = 4; //number of 222 MHz pin
19: const int Pin432 = 5; //number of 432 MHz pin
20: const int Pin902 = 6; //number of 902 MHz pin
21: const int Pin1296 = 8; //number of 1296 MHz pin
22: const int Pin2304 = A5; //number of 2304 MHz pin
23: const int Pin3G = A4; //number of 3GHz pin
24: const int Pin5G = A3; //number of 5GHz pin
25: const int Pin10G = A2; //number of 10GHz pin
26: const int Pin24G = A1; //number of 24GHz pin
27: const int Pin47G = A0; //number of 47GHz pin
28: const int Pin76G = 7; //number of 76GHz pin
29:
30: void setup() {
31:
32: // define GPIO pins as output pins
33: pinMode(Pin50,OUTPUT);
34: pinMode(Pin144,OUTPUT);
35: pinMode(Pin222,OUTPUT);
36: pinMode(Pin432,OUTPUT);
37: pinMode(Pin902,OUTPUT);
38: pinMode(Pin1296,OUTPUT);
39: pinMode(Pin2304,OUTPUT);
40: pinMode(Pin3G,OUTPUT);
41: pinMode(Pin5G,OUTPUT);
42: pinMode(Pin10G,OUTPUT);
43: pinMode(Pin24G,OUTPUT);
44: pinMode(Pin47G,OUTPUT);
45: pinMode(Pin76G,OUTPUT);
46:
47: //initialize all GPIO pin values to low
48: digitalWrite(Pin50,LOW);
49: digitalWrite(Pin144,LOW);
50: digitalWrite(Pin222,LOW);
51: digitalWrite(Pin432,LOW);
52: digitalWrite(Pin902,LOW);
53: digitalWrite(Pin1296,LOW);
54: digitalWrite(Pin2304,LOW);
55: digitalWrite(Pin3G,LOW);
56: digitalWrite(Pin5G,LOW);
57: digitalWrite(Pin10G,LOW);
58: digitalWrite(Pin24G,LOW);
59: digitalWrite(Pin47G,LOW);
60: digitalWrite(Pin76G,LOW);
61:
62: // define, start, flush serial port Serial 0
63: // VHF log will send commands to this port
64: Serial.begin(9600, SERIAL_8N1); // 9600/8/N/1
65: Serial.println("N1MM Bandswitch");
```

```
66: Serial.println("By W3SZ");
67: Serial.println("Uses USB-Serial Port and OTRSP Protocol");
68: Serial.println("50 MHz thru 76 GHz");
69: delay(100);
70:
71: Serial.flush(); // clear buffers
72: }
73:
74: void loop() { //MAIN
75:
76: /////////////////////////////////////////////////////////////////// Get the Command ///////////////////////////////////////////////////////////////////
77: // get VHFLOG command from serial0
78: if (commandStringComplete) {
79:   command = commandInputString;
80:   // save this new command then clear the input buffer
81:   // clear the string:
82:   commandInputString = "";
83:   //set string complete flag to false in preparation for next VHFLOG command;
84:   commandStringComplete = false;
85: }
86: /////////////////////////////////////////////////////////////////// End Command ///////////////////////////////////////////////////////////////////
87: // now process the VHFLOG command
88: if (command.length() > 0){
89: /////////////////////////////////////////////////////////////////// Commands ///////////////////////////////////////////////////////////////////
90:
91:   Serial.print("Command is:");
92:   Serial.print(command);
93:
94:   if ((command.indexOf("AUX100")>=0) || (command.indexOf("AUX200")>=0) ) { //
95:     //set band to 6m
96:     //set Pin50 high, all other pins low
97:     digitalWrite(Pin50,HIGH);
98:     digitalWrite(Pin144,LOW);
99:     digitalWrite(Pin222,LOW);
100:    digitalWrite(Pin432,LOW);
101:    digitalWrite(Pin902,LOW);
102:    digitalWrite(Pin1296,LOW);
103:    digitalWrite(Pin2304,LOW);
104:    digitalWrite(Pin3G,LOW);
105:    digitalWrite(Pin5G,LOW);
106:    digitalWrite(Pin10G,LOW);
107:    digitalWrite(Pin24G,LOW);
108:    digitalWrite(Pin47G,LOW);
109:    digitalWrite(Pin76G,LOW);
110:    Serial.print("Pin50 High");
111:   }
112:   else if ((command.indexOf("AUX101")>=0) || (command.indexOf("AUX201")>=0) ) {
113:     // set band to 2m
114:     //set Pin144 high, all other pins low
115:     digitalWrite(Pin50,LOW);
116:     digitalWrite(Pin144,HIGH);
117:     digitalWrite(Pin222,LOW);
118:     digitalWrite(Pin432,LOW);
119:     digitalWrite(Pin902,LOW);
120:     digitalWrite(Pin1296,LOW);
121:     digitalWrite(Pin2304,LOW);
122:     digitalWrite(Pin3G,LOW);
123:     digitalWrite(Pin5G,LOW);
124:     digitalWrite(Pin10G,LOW);
125:     digitalWrite(Pin24G,LOW);
126:     digitalWrite(Pin47G,LOW);
127:     digitalWrite(Pin76G,LOW);
128:     Serial.print("Pin144 High");
129:   }
```

```
130:     else if ((command.indexOf("AUX102")>=0) || (command.indexOf("AUX202")>=0) ) {
        // set band to 222
131:         //set Pin222 high, all other pins low
132: digitalWrite(Pin50,LOW);
133: digitalWrite(Pin144,LOW);
134: digitalWrite(Pin222,HIGH);
135: digitalWrite(Pin432,LOW);
136: digitalWrite(Pin902,LOW);
137: digitalWrite(Pin1296,LOW);
138: digitalWrite(Pin2304,LOW);
139: digitalWrite(Pin3G,LOW);
140: digitalWrite(Pin5G,LOW);
141: digitalWrite(Pin10G,LOW);
142: digitalWrite(Pin24G,LOW);
143: digitalWrite(Pin47G,LOW);
144: digitalWrite(Pin76G,LOW);
145:     }
146:
147:     else if ((command.indexOf("AUX103")>=0) || (command.indexOf("AUX203")>=0) ) {
        // set band to 432
148:         //set Pin432 high, all other pins low
149: digitalWrite(Pin50,LOW);
150: digitalWrite(Pin144,LOW);
151: digitalWrite(Pin222,LOW);
152: digitalWrite(Pin432,HIGH);
153: digitalWrite(Pin902,LOW);
154: digitalWrite(Pin1296,LOW);
155: digitalWrite(Pin2304,LOW);
156: digitalWrite(Pin3G,LOW);
157: digitalWrite(Pin5G,LOW);
158: digitalWrite(Pin10G,LOW);
159: digitalWrite(Pin24G,LOW);
160: digitalWrite(Pin47G,LOW);
161: digitalWrite(Pin76G,LOW);
162:     }
163:
164:     else if ((command.indexOf("AUX104")>=0) || (command.indexOf("AUX204")>=0) ) {
        // set band to 903
165:         //set Pin902 high, all other pins low
166: digitalWrite(Pin50,LOW);
167: digitalWrite(Pin144,LOW);
168: digitalWrite(Pin222,LOW);
169: digitalWrite(Pin432,LOW);
170: digitalWrite(Pin902,HIGH);
171: digitalWrite(Pin1296,LOW);
172: digitalWrite(Pin2304,LOW);
173: digitalWrite(Pin3G,LOW);
174: digitalWrite(Pin5G,LOW);
175: digitalWrite(Pin10G,LOW);
176: digitalWrite(Pin24G,LOW);
177: digitalWrite(Pin47G,LOW);
178: digitalWrite(Pin76G,LOW);
179:     }
180:
181:     else if ((command.indexOf("AUX105")>=0) || (command.indexOf("AUX205")>=0) ) {
        // set band to 1296
182:         //set Pin1296 high, all other pins low
183: digitalWrite(Pin50,LOW);
184: digitalWrite(Pin144,LOW);
185: digitalWrite(Pin222,LOW);
186: digitalWrite(Pin432,LOW);
187: digitalWrite(Pin902,LOW);
188: digitalWrite(Pin1296,HIGH);
189: digitalWrite(Pin2304,LOW);
190: digitalWrite(Pin3G,LOW);
191: digitalWrite(Pin5G,LOW);
```

```
192: digitalWrite(Pin10G,LOW);
193: digitalWrite(Pin24G,LOW);
194: digitalWrite(Pin47G,LOW);
195: digitalWrite(Pin76G,LOW);
196:     }
197:
198:     else if ((command.indexOf("AUX106")>=0) || (command.indexOf("AUX206")>=0) ) {
199:         // set band to 2304
200:         //set Pin2304 high, all other pins low
201: digitalWrite(Pin50,LOW);
202: digitalWrite(Pin144,LOW);
203: digitalWrite(Pin222,LOW);
204: digitalWrite(Pin432,LOW);
205: digitalWrite(Pin902,LOW);
206: digitalWrite(Pin1296,LOW);
207: digitalWrite(Pin2304,HIGH);
208: digitalWrite(Pin3G,LOW);
209: digitalWrite(Pin5G,LOW);
210: digitalWrite(Pin10G,LOW);
211: digitalWrite(Pin24G,LOW);
212: digitalWrite(Pin47G,LOW);
213: digitalWrite(Pin76G,LOW);
214:     }
215:     else if ((command.indexOf("AUX107")>=0) || (command.indexOf("AUX207")>=0) ) {
216:         // set band to 3456
217:         //set Pin3G high, all other pins low
218: digitalWrite(Pin50,LOW);
219: digitalWrite(Pin144,LOW);
220: digitalWrite(Pin222,LOW);
221: digitalWrite(Pin432,LOW);
222: digitalWrite(Pin902,LOW);
223: digitalWrite(Pin1296,LOW);
224: digitalWrite(Pin2304,LOW);
225: digitalWrite(Pin3G,HIGH);
226: digitalWrite(Pin5G,LOW);
227: digitalWrite(Pin10G,LOW);
228: digitalWrite(Pin24G,LOW);
229: digitalWrite(Pin47G,LOW);
230: digitalWrite(Pin76G,LOW);
231:     }
232:     else if ((command.indexOf("AUX108")>=0) || (command.indexOf("AUX208")>=0) ) {
233:         // set band to 5760
234:         //set Pin5G high, all other pins low
235: digitalWrite(Pin50,LOW);
236: digitalWrite(Pin144,LOW);
237: digitalWrite(Pin222,LOW);
238: digitalWrite(Pin432,LOW);
239: digitalWrite(Pin902,LOW);
240: digitalWrite(Pin1296,LOW);
241: digitalWrite(Pin2304,LOW);
242: digitalWrite(Pin3G,LOW);
243: digitalWrite(Pin5G,HIGH);
244: digitalWrite(Pin10G,LOW);
245: digitalWrite(Pin24G,LOW);
246: digitalWrite(Pin47G,LOW);
247: digitalWrite(Pin76G,LOW);
248:     }
249:     else if ((command.indexOf("AUX109")>=0) || (command.indexOf("AUX209")>=0) ) {
250:         // set band to 10368
251:         //set Pin10G high, all other pins low
252: digitalWrite(Pin50,LOW);
253: digitalWrite(Pin144,LOW);
254: digitalWrite(Pin222,LOW);
```

```
254: digitalWrite(Pin432,LOW);
255: digitalWrite(Pin902,LOW);
256: digitalWrite(Pin1296,LOW);
257: digitalWrite(Pin2304,LOW);
258: digitalWrite(Pin3G,LOW);
259: digitalWrite(Pin5G,LOW);
260: digitalWrite(Pin10G,HIGH);
261: digitalWrite(Pin24G,LOW);
262: digitalWrite(Pin47G,LOW);
263: digitalWrite(Pin76G,LOW);
264:     }
265:
266:     else if ((command.indexOf("AUX110")>=0) || (command.indexOf("AUX210")>=0) ) {
267:         // set band to 24 GHz
268:         //set Pin24G high, all other pins low
269: digitalWrite(Pin50,LOW);
270: digitalWrite(Pin144,LOW);
271: digitalWrite(Pin222,LOW);
272: digitalWrite(Pin432,LOW);
273: digitalWrite(Pin902,LOW);
274: digitalWrite(Pin1296,LOW);
275: digitalWrite(Pin2304,LOW);
276: digitalWrite(Pin3G,LOW);
277: digitalWrite(Pin5G,LOW);
278: digitalWrite(Pin10G,LOW);
279: digitalWrite(Pin24G,HIGH);
280: digitalWrite(Pin47G,LOW);
281: digitalWrite(Pin76G,LOW);
282:     }
283:     else if ((command.indexOf("AUX111")>=0) || (command.indexOf("AUX211")>=0) ) {
284:         // set band 47 GHz
285:         //set Pin47G high, all other pins low
286: digitalWrite(Pin50,LOW);
287: digitalWrite(Pin144,LOW);
288: digitalWrite(Pin222,LOW);
289: digitalWrite(Pin432,LOW);
290: digitalWrite(Pin902,LOW);
291: digitalWrite(Pin1296,LOW);
292: digitalWrite(Pin2304,LOW);
293: digitalWrite(Pin3G,LOW);
294: digitalWrite(Pin5G,LOW);
295: digitalWrite(Pin10G,LOW);
296: digitalWrite(Pin24G,LOW);
297: digitalWrite(Pin47G,HIGH);
298: digitalWrite(Pin76G,LOW);
299:     }
300:     else if ((command.indexOf("AUX112")>=0) || (command.indexOf("AUX212")>=0) ) {
301:         // set band to 76 GHz
302:         //set Pin76G high, all other pins low
303: digitalWrite(Pin50,LOW);
304: digitalWrite(Pin144,LOW);
305: digitalWrite(Pin222,LOW);
306: digitalWrite(Pin432,LOW);
307: digitalWrite(Pin902,LOW);
308: digitalWrite(Pin1296,LOW);
309: digitalWrite(Pin2304,LOW);
310: digitalWrite(Pin3G,LOW);
311: digitalWrite(Pin5G,LOW);
312: digitalWrite(Pin10G,LOW);
313: digitalWrite(Pin24G,LOW);
314: digitalWrite(Pin47G,LOW);
315: digitalWrite(Pin76G,HIGH);
316:     }
317:     // cleanup
```

```
317:     command = ""; // clear the VHFLOG command
318:   }
319:   //////////////////////////////////// END COMMANDS ////////////////////////////////////
320:
321:
322:   delay(25); // long enough for the radio to return its frequency
323:
324: } //END MAIN
325:
326:
327: /*
328:  SerialEvent occurs whenever a new data comes in the
329:  hardware serial RX. This routine is run between each
330:  time loop() runs, so using inside loop can
331:  response. Multiple bytes of data may be available.
332:  */
333: void serialEvent() {
334:
335:   char commandInChar;
336:
337:   while (Serial.available()) { // interrupt generated by hardware serial port
338:     // get the new byte:
339:     commandInChar = (char)Serial.read();
340:
341:     // add it to the commandInputString:
342:     commandInputString += commandInChar; // append
343:     // look for a carriage return
344:     // so the main loop can do something about it:
345:     if (commandInChar == '\r') { // the commands all end with a CR
346:       commandStringComplete = true;
347:     }
348:   }
349: }
350:
351:
352:
```